MAT 160 PROJECT 3 MISSAEL COLIN CUEVAS
CAESAR HERNANDEZ
GURPARKASH SINGH

1. CHALLENGE 1: The goal is for you to understand the idea of *Support Vector Machines* (SVMs) and the power of linear optimization for classification of data. In this project, we will use linear programming for *breast cancer diagnosis*. The project will use the Wisconsin Diagnosis Breast Cancer Database (WDBC). The idea is to come up with a discriminant function (a separating plane in this case) to determine if an unknown sample is benign or malignant. In order to do this, you will use part of the data in the above database as a "training set" to generate your separating plane and the remaining part as a testing set to test your separating plane. Attributes 3 to 32 form a 30-dimensional vector representing each case as a point in 30-dimensional real space $R^{30}$. To generate the separating plane, a training set, consisting of two disjoint point sets $B$ and $M$ in $R^{30}$ representing confirmed benign and malignant cases. The separating plane, to be determined by solving a single linear program in MATLAB or SCIP is based on the formulation proposed in class.

   (a) Formulate the problem as a linear program. Solve the problem using the $M$ and $B$ as a training set from the first 500 cases of the wdbc.data file available from
   `https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)`
   The last 69 points should be used as a testing set. Solve the linear program and print out the separating hyperplane, and the minimum value of the LP.

   (b) Test the separating plane on the 69 cases of the testing set. Report the number of misclassified points on the testing set. It is probably a good idea if you create an MATLAB file to do this.

   (c) We saw in class that there is a a *quadratic programming* formulation of the separation problem (to maximize margin distance). Formulate the quadratic program in SCIP and try to solve it that way. Is the separating hyperplane better?

**As It would be shown in the code below, the program will use 5 different values for lambda. It prints the hyperplane and minimum value of the LP for each lambda. It also prints the misclassified points on the testing set and I used cvx instead of SCIP. *Used ReadWDBC to extract wdbc.data.**

```
clc;
clear all;
close all;
[train,test,ntrain,ntest]=ReadWDBC('wdbc.data',30);
Malignant=[];
Benign=[];
T = [];
M = [];
MinValue = [];
for i = 1:ntrain
    if(train(i,1)==0)
        Benign = [Benign;train(i,2:end)]; %Create Benign
    elseif(train(i,1)==1)
        Malignant = [Malignant;train(i,2:end)]; %Create malignant.
    end
end
m = size(Malignant,1);
k = size(Benign,1);
e_m = ones(m,1);%unit e_m vector
e_k = ones(k,1);%unit e_k vector
lambda = [0.001 0.01 0.1 1 10];
for i = 1:5
    l = lambda(i);
cvx_begin quiet%optimize SVM classification
    variable w(30)
    variable Y(1)
    variable y(m)%binary M
    variable z(k)%binary B
    minimize((1/m)*e_m'*y + (1/k)*e_k'*z + (l/2)*w'*w)
    subject to
    Malignant*w - e_m*Y +y >= e_m
```

```
    -Benign*w + e_k*Y+z >= e_k
    y >= 0
    z >= 0
cvx_end
T(:,i) = w; %make table take hyperplane
U(i) = Y; %make table take Y
MinValue(i) = cvx_optval;
end
BenignT = [];
MalignantT = [];
for i = 1:ntest
    if(test(i,1)==0)
        BenignT = [BenignT;test(i,2:end)];
    elseif(test(i,1)==1)
        MalignantT = [MalignantT;test(i,2:end)];
    end
end
for j = 1:5
MisClassifiedPoints = 0;
ma = size(MalignantT,1);
be = size(BenignT,1);
e_mm = ones(ma,1);
e_kk = ones(be,1);
w = T(:,j);%find mismatches for each hyperplane
Y = U(j);
a = MalignantT*w - e_mm *Y;
b = BenignT*w - e_kk *Y;
for i = 1:ma
    if(a(i) <= 0)
        MisClassifiedPoints = MisClassifiedPoints+1;
    end
end
for i = 1:be
    if(b(i)>0)
        MisClassifiedPoints = MisClassifiedPoints+1;
    end
end
M(j) = MisClassifiedPoints;%store mismatches
end
colNames = {'sigma0_001','sigma0_01','sigma0_1','sigma_1','sigma10'};
sTable = array2table(T,'VariableNames',colNames);
MisClassified_Points = array2table(M,'VariableNames',colNames);
Optimal_Value_LP = array2table(MinValue,'VariableNames',colNames);
display(sTable);
display(Optimal_Value_LP);
display(MisClassified_Points);
```

**Function ReadWDBC**

```
function [train,test,ntrain,ntest] = ReadWDBC(datafile,dataDim)
fp = fopen(datafile,'r');
samples = 0;
train = zeros(0,dataDim+1);
[id,count] = fscanf(fp,'%d,',1);
while (count == 1) %Read if B or M and Convert to 1 and 0
  samples = samples + 1;
  type = fscanf(fp,'%1s',1);
  if type=='B'
    type =0;
  elseif type=='M'
    type =1;
  end
  vec = fscanf(fp,',%e',dataDim);
  train = [train; type vec'];
  [id,count] = fscanf(fp,'%d,',1);
end
test  = train(501:samples,:);%Create size 69 testing Set
train = train(1:500,:);%Create size 500 training Set
ntrain = size(train);
ntest = size(test);
return;
```

sTable =

30×5 table

| sigma0_001 | sigma0_01 | sigma0_1 | sigma_1 | sigma10 |
|---|---|---|---|---|
| -1.7211 | -0.51729 | -0.11442 | -0.0096883 | -0.0009455 |
| -0.052152 | -0.070678 | -0.030024 | 0.026011 | 0.025007 |
| 0.15427 | 0.213 | 0.10239 | 0.064952 | 0.023139 |
| 0.0034145 | -0.018233 | -0.018787 | -0.016559 | -0.012897 |
| 0.80674 | 0.12223 | 0.018344 | 0.0021504 | 0.00029583 |
| 0.32848 | 0.13281 | 0.031818 | 0.0043051 | 0.00093041 |
| 0.8221 | 0.25606 | 0.049767 | 0.0065016 | 0.0012965 |
| 1.1349 | 0.17425 | 0.026145 | 0.00316 | 0.00053221 |
| 0.98729 | 0.17495 | 0.02077 | 0.0025994 | 0.00044029 |
| 0.0089009 | 0.016237 | 0.0050301 | 0.00062562 | 0.00011834 |
| 0.39015 | 0.045885 | 0.0067492 | 0.0005735 | 0.00021828 |
| -1.5886 | -0.71062 | -0.17914 | -0.015501 | -0.0002597 |
| 0.34939 | -0.15323 | -0.056304 | 0.0021244 | 0.0035201 |
| 0.050296 | 0.070148 | 0.058418 | 0.043752 | 0.025109 |
| 0.11098 | 0.013893 | 0.0025388 | 0.00032244 | 3.9655e-05 |
| -0.73929 | -0.050929 | 0.001243 | 0.00047294 | 0.00023309 |
| -0.72197 | -0.0097322 | 0.0067264 | 0.0011669 | 0.0003235 |
| 0.16057 | 0.024666 | 0.0038849 | 0.00052646 | 8.2777e-05 |
| 0.086571 | 0.0097739 | 0.0033226 | 0.00035583 | 8.5329e-05 |
| -0.11811 | -0.010083 | -0.00017731 | 2.1476e-05 | 1.8788e-05 |
| 0.43918 | -0.019703 | 0.013379 | 0.0021998 | 0.00024023 |
| 0.28276 | 0.25042 | 0.19576 | 0.096768 | 0.041974 |
| -0.019592 | 0.055553 | 0.15693 | 0.11329 | 0.049217 |
| 0.0044277 | 0.0079798 | 0.0053618 | 0.007266 | 0.010559 |
| 1.6646 | 0.25407 | 0.036201 | 0.0042177 | 0.00054762 |
| 0.15703 | 0.49407 | 0.11346 | 0.014819 | 0.0032568 |
| 2.0257 | 0.87861 | 0.15355 | 0.019242 | 0.0039413 |
| 2.5422 | 0.40975 | 0.057877 | 0.0070784 | 0.0011158 |
| 2.4031 | 0.4194 | 0.065388 | 0.0072676 | 0.0011841 |
| 0.10023 | 0.064493 | 0.015132 | 0.0019176 | 0.00037114 |

Optimal_Value_LP =

1×5 table

| sigma0_001 | sigma0_01 | sigma0_1 | sigma_1 | sigma10 |
|---|---|---|---|---|
| 0.17002 | 0.20302 | 0.22779 | 0.25212 | 0.31208 |

```
MisClassified_Points =

  1×5 table

    sigma0_001    sigma0_01    sigma0_1    sigma_1    sigma10
    __________    _________    ________    _______    _______

        2             2            2          4          4

>>
```

2. PROJECT 2: Finding out the top features that determine what is an "spam email"

   You will write an algorithm in SCIP for deciding what are the most important features that distinguish spam email from regular truthful email. You can download the data files from `http://archive.ics.uci.edu/ml/machine-learning-databases/spambase/`

   In there you can find the data file spambase.data that contains information on 4601 emails (1813 are Spam!!) Each row contains 58 attributes: 58 (57 continuous, 1 nominal class label). The last attribute in the very last column of 'spambase.data' denotes whether the e-mail was considered spam (1) or not (0).

   Most of the attributes indicate whether a particular word or character was frequently occurring in the e-mail. The run-length attributes (55-57) measure the length of sequences of consecutive capital letters. For more information see the documentation.

   **GOAL** use the LASSO method to figure out what are the most significant of the characteristics that define the spam emails (e.g., maybe number of consecutive capital letters is overly excessive?). Can you run LASSO directly? You can rewrite the LASSO convex model as a linear program by adding some extra variables!

**In the code and problem below, we used 5 different values for lambda to interpret the date. I rearrange the table according to the characteristics that define the spam emails. As shown in the table, the word "you" is frequently used in spam emails, also, capital letters are frequently used that's why they did not approach faster 0 even though we increased lambda.**

```
clc;
clear all;
close all;
x = load('spambase.data');
names = fopen('namesspam.txt','r');
dataArray = textscan(names,'%s%[^\n\r]');
namesspan = dataArray(:,1);
Y = x(:,58); %read nominal class label
X = x(:,1:57); %read attributes
[m,n] = size(X);
T = [];
lambda =1;
for i = 1:5
cvx_begin quiet% LASSO implementation
    variable w(n);
    minimize norm(Y-X*w,2) + lambda * norm(w,1);
    %subject to
    %norm(w,1) <= lambda;  (LASSO (2) example I implemented 1)
cvx_end
T(:,i) = w;
lambda = lambda + 2*i; %Try diffent values for lambda
end

Z = abs(T);
for i= 1:n
 Rowsum(i,1) = sum(Z(i,:));
end
T(:,6) = Rowsum;
Total = array2table(T,...
    'VariableNames',{'lamda1','lamda3','lamda5','lamda7','lamda9','AbsSum'},...
    'RowNames',namesspan{:});
Total = sortrows(Total,6, 'descend')
```

```
Total =

  57×6 table
```

| | lamda1 | lamda3 | lamda5 | lamda7 | lamda9 | AbsSum |
|---|---|---|---|---|---|---|
| word_freq_remove | 0.22371 | 0.18336 | 0.066914 | 4.0655e-09 | 9.1816e-10 | 0.47399 |
| word_freq_your | 0.074991 | 0.091794 | 0.11055 | 0.10414 | 0.079787 | 0.46126 |
| word_freq_000 | 0.17756 | 0.15395 | 1.014e-07 | 1.9697e-09 | 5.9462e-10 | 0.33151 |
| word_freq_you | 0.039226 | 0.048933 | 0.064364 | 0.076069 | 0.077073 | 0.30567 |
| word_freq_free | 0.087337 | 0.085466 | 0.075073 | 0.036096 | 5.6628e-09 | 0.28397 |
| word_freq_our | 0.098798 | 0.088651 | 0.066897 | 0.0018969 | 2.2527e-09 | 0.25624 |
| char_freq_$ | 0.18549 | 0.046491 | 3.745e-09 | 8.7207e-10 | 3.5162e-10 | 0.23198 |
| char_freq_! | 0.074611 | 0.072759 | 0.052648 | 0.0039792 | 2.31e-09 | 0.204 |
| word_freq_internet | 0.10096 | 0.061235 | 7.3203e-09 | 1.3064e-09 | 4.9278e-10 | 0.16219 |
| word_freq_over | 0.11642 | 0.020001 | 3.3852e-09 | 8.4171e-10 | 3.4886e-10 | 0.13642 |
| word_freq_email | 0.060605 | 0.045231 | 1.2337e-08 | 2.1022e-09 | 6.8987e-10 | 0.10584 |
| word_freq_credit | 0.060401 | 0.03428 | 5.3406e-09 | 1.1437e-09 | 4.4249e-10 | 0.094681 |
| word_freq_money | 0.067103 | 0.02589 | 2.9552e-09 | 7.918e-10 | 3.6705e-10 | 0.092993 |
| word_freq_all | 0.057279 | 0.034932 | 7.4614e-09 | 1.9485e-09 | 6.9095e-10 | 0.092211 |
| word_freq_business | 0.053541 | 0.038359 | 6.02e-09 | 2.0092e-09 | 6.5773e-10 | 0.0919 |
| word_freq_font | 0.039218 | 0.027877 | 0.0045186 | 1.5747e-09 | 5.5763e-10 | 0.071613 |
| word_freq_order | 0.045521 | 2.3929e-09 | 1.6742e-09 | 5.2998e-10 | 2.4612e-10 | 0.045522 |
| word_freq_re | -0.020889 | -0.015015 | -4.1827e-09 | -8.1031e-10 | -2.6622e-10 | 0.035904 |
| word_freq_hp | -0.013135 | -0.014107 | -0.0070904 | -1.2416e-09 | -7.3326e-10 | 0.034333 |
| word_freq_mail | 0.020909 | 0.0091357 | 3.9166e-09 | 1.2358e-09 | 5.0654e-10 | 0.030044 |
| word_freq_meeting | -0.022255 | -0.0055516 | -1.481e-09 | -2.9007e-10 | -1.3287e-10 | 0.027807 |
| word_freq_edu | -0.016997 | -0.0076538 | -2.3798e-09 | -7.5467e-10 | -2.8311e-10 | 0.02465 |
| word_freq_3d | 0.012618 | 0.0082565 | 2.1289e-09 | 9.2667e-10 | 4.4132e-10 | 0.020874 |
| word_freq_hpl | -0.010933 | -2.3088e-06 | -2.3274e-09 | -7.094e-10 | -2.7795e-10 | 0.010936 |
| char_freq_; | -0.0095676 | -3.668e-10 | -2.2645e-10 | -7.983e-11 | -2.9056e-11 | 0.0095676 |
| word_freq_people | 0.0067256 | 1.2902e-09 | 1.1754e-09 | 4.1734e-10 | 1.9989e-10 | 0.0067256 |

| | | | | | | |
|---|---|---|---|---|---|---|
| word_freq_george | -0.0028574 | -0.002577 | -0.001135 | -1.4316e-09 | -4.2268e-10 | 0.0065695 |
| char_freq_# | 0.0044887 | 7.3904e-10 | 9.0437e-10 | 3.31e-10 | 1.5596e-10 | 0.0044887 |
| word_freq_data | -0.0039653 | -5.3322e-10 | -6.3851e-10 | -2.0454e-10 | -9.3094e-11 | 0.0039653 |
| word_freq_project | -0.0031432 | -4.6041e-10 | -4.9633e-10 | -1.3546e-10 | -6.0651e-11 | 0.0031432 |
| capital_run_length_average | 0.00050265 | 0.00053354 | 0.00056329 | 0.00069284 | 0.00069288 | 0.0029852 |
| capital_run_length_total | 0.00013188 | 0.00016594 | 0.00020856 | 0.00022484 | 0.00024148 | 0.0009727 |
| capital_run_length_longest | 2.4914e-05 | 8.6127e-05 | 0.00017019 | 0.0002158 | 0.00023939 | 0.00073641 |
| word_freq_pm | -0.00018993 | -5.4368e-10 | -6.0867e-10 | -1.9992e-10 | -8.2205e-11 | 0.00018993 |
| word_freq_1999 | -8.9346e-08 | -5.1112e-10 | -5.4903e-10 | -1.7159e-10 | -7.0149e-11 | 9.0648e-08 |
| word_freq_85 | -7.1037e-08 | -4.7015e-10 | -6.5405e-10 | -2.5706e-10 | -1.0437e-10 | 7.2523e-08 |
| word_freq_labs | -5.1678e-08 | -3.9903e-10 | -6.1004e-10 | -2.3724e-10 | -9.2072e-11 | 5.3016e-08 |
| word_freq_address | 3.1765e-08 | 9.0457e-11 | 5.5998e-10 | 3.5459e-10 | 1.9832e-10 | 3.2968e-08 |
| word_freq_technology | 2.9235e-08 | 1.2978e-10 | 6.4564e-11 | 1.3322e-11 | 1.1196e-11 | 2.9454e-08 |
| word_freq_original | -2.2885e-08 | -2.8322e-10 | -3.78e-10 | -1.3382e-10 | -5.3447e-11 | 2.3734e-08 |
| word_freq_addresses | 1.9517e-08 | 7.9387e-10 | 1.4407e-09 | 4.4115e-10 | 1.9736e-10 | 2.2391e-08 |
| word_freq_conference | -1.863e-08 | -2.3579e-10 | -3.024e-10 | -1.1054e-10 | -4.6622e-11 | 1.9325e-08 |
| word_freq_report | 1.3803e-08 | 2.0805e-10 | 2.2382e-10 | 9.8087e-11 | 5.4539e-11 | 1.4387e-08 |
| word_freq_parts | -1.3623e-08 | -1.897e-10 | -1.3402e-10 | -1.2018e-11 | -2.3781e-12 | 1.3961e-08 |
| word_freq_receive | 1.2074e-08 | 3.2081e-10 | 6.8002e-10 | 3.0994e-10 | 1.693e-10 | 1.3554e-08 |
| word_freq_direct | 1.0166e-08 | 8.0539e-11 | 9.324e-11 | 3.7598e-11 | 2.8479e-11 | 1.0406e-08 |
| word_freq_lab | -8.5624e-09 | -6.1129e-10 | -7.4091e-10 | -2.252e-10 | -9.6266e-11 | 1.0236e-08 |
| word_freq_cs | -8.9237e-09 | -2.6646e-10 | -3.3034e-10 | -1.1965e-10 | -5.499e-11 | 9.6952e-09 |
| char_freq_( | 9.3162e-09 | 6.8135e-11 | 3.5348e-11 | 4.0174e-11 | 2.8108e-11 | 9.488e-09 |
| word_freq_will | -4.286e-09 | 1.2325e-10 | 7.9396e-10 | 7.7153e-10 | 4.7254e-10 | 6.4473e-09 |
| word_freq_telnet | -5.564e-09 | -1.7156e-10 | -3.33e-10 | -1.3982e-10 | -5.5704e-11 | 6.2641e-09 |
| word_freq_make | -5.1541e-09 | 1.0795e-10 | 4.2021e-10 | 2.3262e-10 | 1.5055e-10 | 6.0654e-09 |
| word_freq_table | -4.6941e-09 | -6.5979e-11 | -8.8045e-11 | -3.2302e-11 | -1.2278e-11 | 4.8927e-09 |
| word_freq_650 | 1.5477e-09 | -1.2799e-10 | -3.4024e-10 | -1.5095e-10 | -5.8232e-11 | 2.2251e-09 |
| word_freq_415 | 1.2908e-09 | -7.97e-11 | -2.458e-10 | -9.9187e-11 | -3.9842e-11 | 1.7554e-09 |
| word_freq_857 | 9.169e-10 | -8.3884e-11 | -2.5005e-10 | -1.0106e-10 | -4.0521e-11 | 1.3924e-09 |
| char_freq_[ | -3.6203e-10 | -1.2725e-11 | -9.7244e-12 | -1.3672e-12 | 2.1403e-13 | 3.8606e-10 |

>>

**CHALLENGE 3: You need to write a SCIP model to solve the following geometric problem: Your problem is packing m of spheres in a box of minimal area. The spheres have a given radius ri , and the problem is to determine the precise location of the centers xi . The constraints in this problem are that the spheres should not overlap, and should be contained in a square of center 0 and half-size R. The objective is to minimize the area of the containing box. • Show that two spheres of radius r1, r2 and centers x1, x2 respectively do not intersect if and only if ||x1 − x2||2 exceeds a certain number, which you will determine. • Formulate the sphere packing problem as an optimization model. Is the formulation you have found convex optimization? • Using your model write SCIP code to solve the packing problem of five and six circular disks of the same radius inside a square of half-size R. What is the optimal size if the disks have radius 1? Do some drawings using MATLAB of the packings you discovered. Is the solution unique?**

You need to pack m spheres in a box of minimal area. The spheres have a given radius $r_i$, and the problem is to determine the precise location of centers $x_i$. Constraints:

- spheres do not overlap
- square centered at 0 and half-size R
- objective: minimize area of containing box.

- Show that two sheres of radius $r_1, r_2$ and centers $x_1, x_2$ respectively do not intersect iff $\|x_1 - x_2\|_2$ exceed a certain number, which you will determine.

We need to make sure the spheres do not overlap, this condition is if

$$\|x_1 - x_2\|_2 \geq r_1 + r_2$$

- Formulate sphere packing problem as an optimization model. Is the formulation you have found convex optimization?

Objective minimize: R

$$\|x_i - x_j\|_2 \geq r_i + r_j, \quad \text{for all } i \geq 1, m \geq j$$

$$\|x_i\|_\infty + r_i \leq R, \quad i = 1, \ldots, m,$$

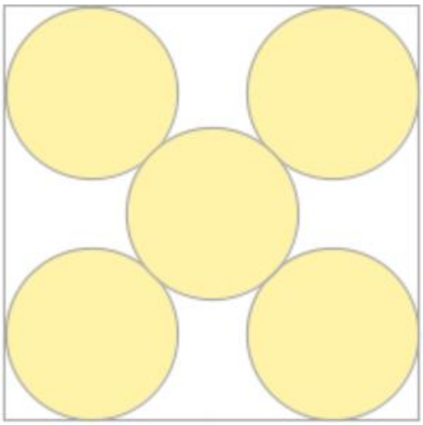
No, this is not a convex optimization because the non-overlapping constraints are bounded from bellow.

**Is the solution unique?**

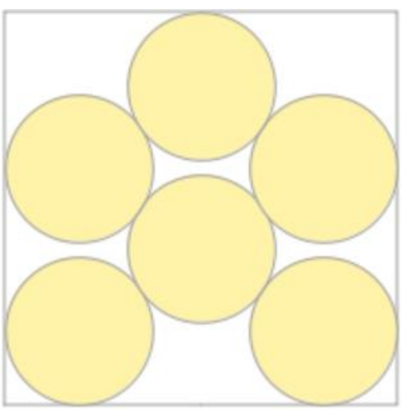
The location of the disks is unique for 5 circles at it is symmetric but it is not unique for the 6 circles as the location of the solutions can be reflected and rotated, having different locations of disks.

For 6 disks, the optimal length is 5.46 and for 5 disks the optimal length is 4.83.

**Packing 5 disks in a circle: (via wolfram)**



**Packing 6 disks in a circle:**



SCIP CODE for Optimization

```
param spheres := 5;

set C := {1 .. spheres};

set P := {<i, j> in C * C with i < j };

#coordinates of squares

var R <= 10;

var x[C] ;

var y[C];

#minimize R
```

```
minimize R: R;

#sphere is in bounds of square

subto c1: forall <i> in C: -R + 1 <= x[i];

subto c2: forall <i> in C: x[i] <= R - 1 ;

subto c3: forall <i> in C: -R + 1  <= y[i];

subto c4: forall <i> in C: y[i] <= R - 1 ;

#the spheres not overlapping

subto c5: forall <i,j> in P do sqrt((x[i] - x[j])^2 + (y[i] - y[j])^2)  >= 2;
```

#4.) Suppose $C \neq \emptyset$ and $C \subseteq \mathbb{R}^n$ and $\lambda_1, \lambda_2 \in \mathbb{R}$

Since $C$ is convex, $C$ consist of $\forall x_1, x_2 \in C$

and $\theta \in [0,1]$ such that $(\theta x_1 + (1-\theta)x_2) \in C$

Suppose $(\lambda_1 + \lambda_2) C$, by scalar mult

$(\lambda_1 + \lambda_2)\theta x_1 + (\lambda_1 + \lambda_2)(1-\theta)x_2$

$\lambda_1 \theta x_1 + \lambda_2 \theta x_2 + \lambda_1 (1-\theta)x_2 + \lambda_2 (1-\theta)x_2$

$\lambda_1 \theta x_1 + \lambda_1 (1-\theta)x_2 + \lambda_2 \theta x_2 + \lambda_2 (1-\theta)x_2$

$\lambda_1 C + \lambda_2 C$

Not convex set $S = \{ s \mid s = \sin(\theta),\ \theta \in \mathbb{R} \}$ since line from $[\sin(0), \sin(\pi)] \cap S = \emptyset$

- $\max_{a>0} \{ a(x+y) - y \cdot a \ln(a) \} = \max_{a>0} \{ f(a) \}$

$\frac{\partial f}{\partial a} = (x+y) - y[\ln(a) + a \cdot 1/|a|] := 0$

$= (x+y) - y\ln(a) - y$

$\Rightarrow x = y\ln(a) \Leftrightarrow e^x = a^y \Rightarrow a = e^{x/y}$

By def of convexity

$f: \mathbb{R}^n \to \mathbb{R}$ is convex if $\forall x, y \in \mathbb{R}^n$

and $\theta \in [0,1]$ we have

$f(\theta x + (1-\theta)y) \leq \theta f(x) + (1-\theta) f(y)$

Since $x, y$ are positive real scalars

$f: (0,\infty) \to \mathbb{R}$ let $z_1, z_2 \in (0,\infty) \wedge \theta \in [0,1]$

$f: z_1 e^{z_2}$

$f(\theta z_1 + (1-\theta)z_2) = (\theta z_1 + (1-\theta)z_2) e^{(\theta z_1 + (1-\theta)z_2)} = \theta z_1 e^{\theta z_1} \cdot e^{(1-\theta)z_2} + (1-\theta)z_2 e^{\theta z_1} e^{(1-\theta)z_2}$

$= f(\theta z_1) \cdot e^{(1-\theta)z_2} + f((1-\theta)z_2) \cdot e^{\theta z_1}$

By properties of Exp $\leq \theta f(z_1) + (1-\theta) f(z_2) \quad \forall z_1, z_2 \in (0,\infty)$

Thus $f$ is convex in $(0,\infty)^2$ if $z_2$ dummy var

- $\ln(\sum^n e^{x_i})$ convex?

$e^{x_i}$ is convex since $y_1, y_2 \in \mathbb{R}$ $\theta \in [0,1]$

Use Thm from class

$f$ is convex $\Leftrightarrow$ Twice differentiable on convex domain $(ax)$

$\frac{\partial}{\partial x}(e^{ax}) = a e^{ax}$ ; $\frac{\partial^2}{\partial x^2}\{e^{ax}\} = a^2 e^{ax}$ ; $a^2 \in [0,\infty]$

thus $\nabla^2 f \geq 0 \quad \forall z \in \{ax \mid x \in \mathbb{R}\}$

Now that $e^{x_i}$ is established to be convex

We can describe $e^{x_i}$ as the set $\{y \mid y = e^{x} \text{ where } x \in \mathbb{R}\}$

By properties of convex sets

if $e^{x_1} = C$ and $e^{x_2} = D$ both convex

then $C \cap D$ is convex $\Leftrightarrow \{z \mid y : e^{x_1} + g : e^{x_2}, x_1, x_2 \in \mathbb{R}\}$

By induction $\bigcap_{i=1}^{n} e^{x_i}$ is convex

$\frac{\partial^2}{\partial x^2} \ln(x) = -1/x^2$

Since $\ln(x)$ is twice differentiable and domain$(\ln) = \bigcap_{i=1}^{n} e^{x_i}$ is convex $\Rightarrow$ $\ln(x)$ is convex (strictly concave since negative for all values $\nabla^2 f$)

- $S_k : \mathbb{R}^n \to \mathbb{R}$ such that $S_k(x) = \sum_{i=1}^{k} x_{[i]}$ $x_{[i]}$ max componet

Let $z_1, z_2 \in \mathbb{R}^n$ now let $l := $ max component of $z_1$ and $q :=$ max component of $z_2$

$S_k(z_1) = k \cdot l$ and $S_k(z_2) = k \cdot q$. Let $\theta \in [0,1]$

$$\begin{aligned} S_k(\theta z_1 + (1-\theta) z_2) &= \sum_1^k (\theta z_{1[i]} + (1-\theta) z_{2[i]}) \\ &= \theta \sum_1^k z_{1[i]} + (1-\theta) \sum_i^k z_{2[i]} \\ &\leq \theta \cdot k \cdot l + (1-\theta) \cdot kq = \theta S_k(z_1) + (1-\theta) S_k(z_2) \end{aligned}$$

Thus $S_k(x)$ is convex.

- $\phi : \mathbb{R}^n \to \mathbb{R}$ s.t. $\phi(x) = 1/n \sum_{i:\text{odd in}} |x_i - \text{med}(x)|$

Want to show $g(x) = |x_i - \text{md}(x)| \subseteq S_k(x)$

Note $\max\{\sum |x_i - \text{med}(x)|\} = n \cdot |S_k(x_i) - \text{md}(x)|$

then by Cauchy-Schwartz $|S_k(x) - 0| \leq |S_k(x) + \text{med}(x)| + |\text{med}(x) + 0|$

thus $n \cdot |x_i - \text{med}(x)| \leq n \cdot S_k(x)$.

Therefore by property of convex set $\therefore$ $\phi(x)$ is convex

- Let $F$ be compact in $\mathbb{R}^n$

Since $F$ is compact every Cauchy sequence in $F$ converges in $F$

Let $u \in F$, then $\forall \varepsilon > 0 \; \exists N \in \mathbb{N}$ such that $m, n > N \Rightarrow |s_n - s_m| < \varepsilon$

let the $\lim s_n \to u$

Thus $\forall u \in F$, $u$ is "closest" to itself.

- $\min \{ 2x_1 \arctan(x_1) - \ln(x_1^2+1) + x_2^4 + (x_3-1)^2 \}$
  subject to : $x_1^2 + x_2^2 + x_3^2 - 4 \leq 0$ , $x_1 \geq 0$

  $\ln(x_1^2+1)$ has already been shown to be convex

  $x_2^4$ , $(x_3-1)^2$ are trivially convex

  $f = 2x_1 \arctan(x_1)$ ; $\partial^2 f / \partial x^2 = -\frac{2x}{(1+x^2)^2}$

  thus thus $f$ is convex

  $\sum x_i^2 \leq 4$ for $i : 1, 2, 3$ is convex

  since $x_i^2 : \mathbb{R} \to \mathbb{R}$ $\forall z_1, z_2 \in \mathbb{R}$

  $\nabla^2 x_i^2 = 2 \geq 0$ $\forall z_1, z_2 \in \mathbb{R}$

  Once doing the calculation we get min of

  $f : -0.177931$ , $x_1 = -2.3702 \times 10^{-12}$ , $x_2 = -0.605001$ , $x_3 = 1$

- Consider the Optimization

  $\min \{ x_3 \}$

  subject to : $(x_1-1)^2 + x_2^2 + x_3^2 - 1 \geq 0$

  $(x_1+1)^2 + x_2^2 + x_3^2 - 1 \geq 0$

  $x_1^2 + x_2^2 + x_3^2 - 4 \leq 0$

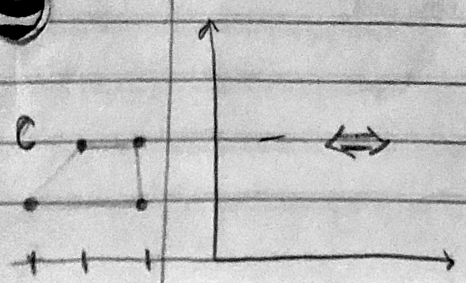
  Given the 3rd constraint intersection with the rest, although convex in $\mathbb{R}^3$ the region is not feasible

  After calculation we get min of

  $x_3 = -2$, where $x_1 = -1.23376 \times 10^{-8}$ ; $x_2 = -7.99124 \times 10^{-9}$ ; $x_3 = -2$

- find max value function $f(x,y) = x^2 + y^4 + xy$

  inside convex hull of $(-1,1), (-1,2), (-2,2), (-3,1)$



$1 \leq y \leq 2$
$x \leq -1$ $\Longleftrightarrow$
$y \leq x + 4$

for min over $C$ : $f = 1$ st $x = -1, y = 1$

for max over $C$ : $f = 16$ st $x = -2, y = 2$

Since $\nabla^2 f = \begin{bmatrix} 2 & 1 \\ 1 & 12y^2 \end{bmatrix}$ $\Rightarrow$ Global Min on $\mathbb{R}^2$ is same as local min

- Let $y \in \mathbb{R}^n$ and $d_y(x) = \|x - y\|_2^2$, $x \in \mathbb{R}^n$

Select a pair of points in $\mathbb{R}^n$

$z_1, z_2 \in \mathbb{R}^n$ such that $z_1 \neq z_2$ and $\lambda \in (0,1)$

$f(\lambda z_1 + (1-\lambda) z_2) = \|(\lambda z_1 + (1-\lambda) z_2) - y\|_2^2$

$= \left(\sqrt{[\lambda z_1 + (1-\lambda) z_2 - y]^2}\right)^2$

Note for the max $|z_i - z_{2i}|$ component of $z_i$ & $z_{2i}$

the metric $< K |z_i - z_{2i}|$ max $i$

thus $\left(\sqrt{[\lambda z_1 + (1-\lambda z_2) - y]^2}\right)^2 =$

$< K |\lambda z_i - (1-\lambda) z_{2i}|$ max $i$

$< \lambda f(z_1) + (1-\lambda) f(z_2)$

Since $z_1 \neq z_2$ this holds for all $z_1, z_2 \in \mathbb{R}^n$

for some fixed $n$

- Let the polyhedron $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ for $A \in \mathbb{R}^{m \times n}$

and $b \in \mathbb{R}^m$

$\max_{x \in P} c^T x$

- Def: $P^0 \stackrel{d}{=} \{d \in \mathbb{R}^n \mid \forall x \in P, \forall \lambda \geq 0, x + \lambda d \in P\}$

fix $d^* \in P^0$, then $\forall x \in P$ $x + \lambda d^* \in P$

- Then $A(x + \lambda d^*) \leq b \Leftrightarrow Ax + A(\lambda d^*) \leq b \Leftrightarrow A\lambda d^* \leq b - \underbrace{Ax}_{\leq b} = 0$

Thus $Ad^* \leq 0$ since $\forall \lambda \geq 0$

By the properties of convex set

the subset of $P^0$ is also convex in $P$

- Suppose $\exists d^* \in P^0$ such that $c^T d^* > 0$

for the objective function $c^T d^*$ max

Thus $c^T d^* \notin P^0$

then $\exists k \in \mathbb{N}$ st $c_k$, the $k^{th}$ vector of $c^T$

such that $Ax + A(\lambda d_i^*) > b$

Since $d_i^*$ lies outside of the $P^0$

then the $c^T x$ is not bounded

in the $k^{th}$ dimension.