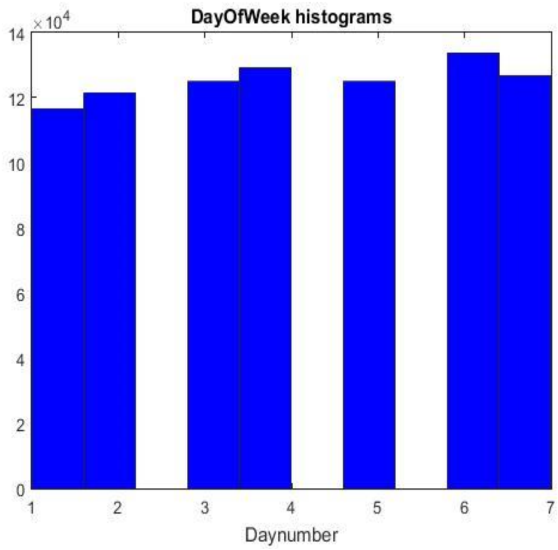
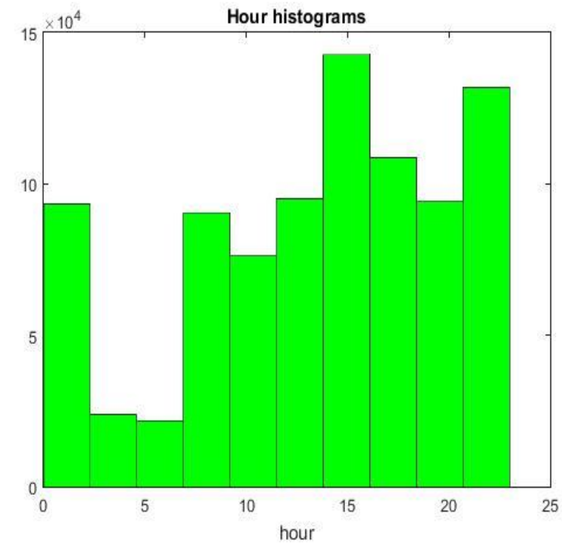
EC503: Learning From Data
Matlab Assignment 2.1
March 18, 2016

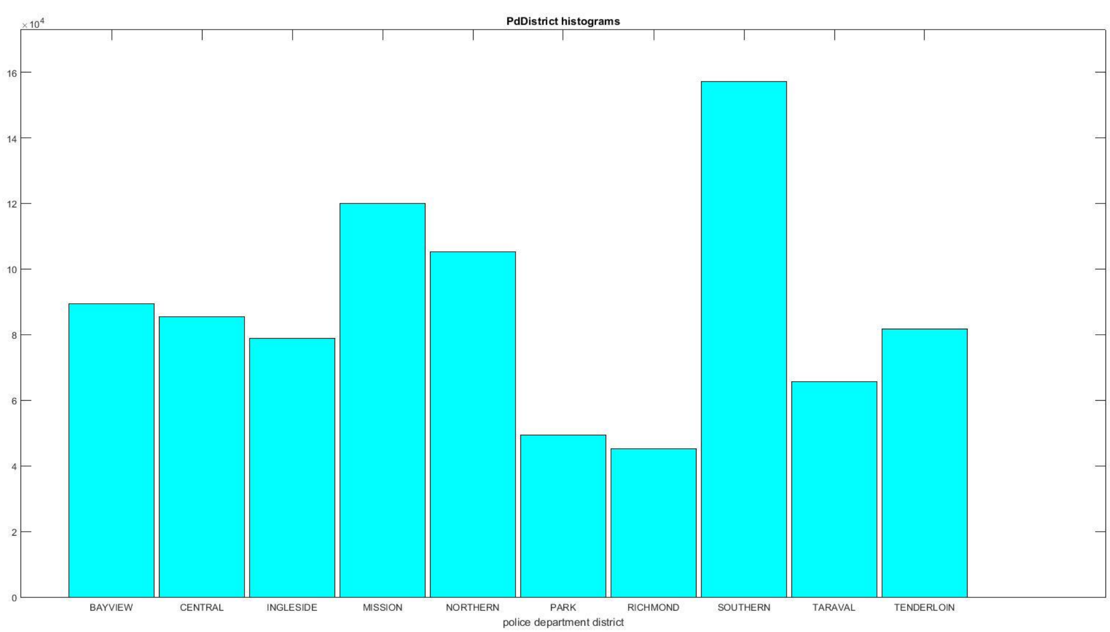
Athar Roshandelpoor
U46981822

Problem 1 part a

ii) Histograms







Athar Roshandelpoor
U46981822

iii)most likely hour of occurrence of each type of crime

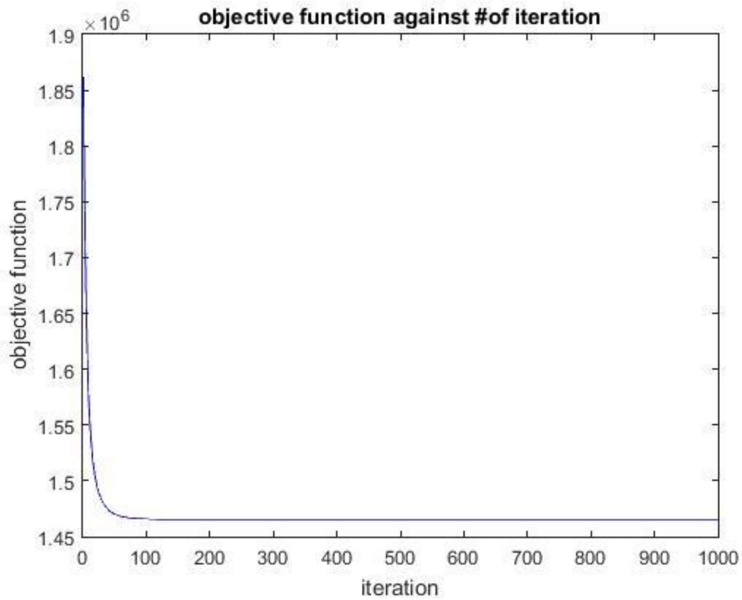
'ARSON'	0
'ASSAULT'	0
'BAD CHECKS'	12
'BRIBERY'	17
'BURGLARY'	17
'DISORDERLY CONDUCT'	6
'DRIVING UNDER THE INFLUENCE'	0
'DRUG/NARCOTIC'	14
'DRUNKENNESS'	0
'EMBEZZLEMENT'	0
'EXTORTION'	0
'FAMILY OFFENSES'	15
'FORGERY/COUNTERFEITING'	0
'FRAUD'	0
'GAMBLING'	13
'KIDNAPPING'	0
'LARCENY/THEFT'	18
'LIQUOR LAWS'	17
'LOITERING'	17
'MISSING PERSON'	8
'NON-CRIMINAL'	12
'OTHER OFFENSES'	17
'PORNOGRAPHY/OBSCENE MAT'	14
'PROSTITUTION'	22
'RECOVERED VEHICLE'	12
'ROBBERY'	21
'RUNAWAY'	18
'SECONDARY CODES'	12
'SEX OFFENSES FORCIBLE'	0
'SEX OFFENSES NON FORCIBLE'	0
'STOLEN PROPERTY'	16
'SUICIDE'	18
'SUSPICIOUS OCC'	12
'TREA'	5
'TRESPASS'	6
'VANDALISM'	18
'VEHICLE THEFT'	18
'ARSON'	0
'ASSAULT'	0

iv)most likely type of crime within each pdDistrict

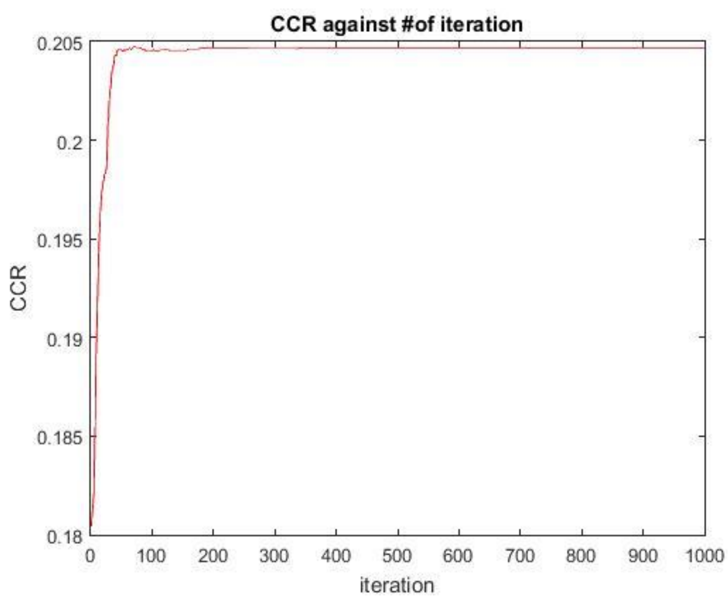
'BAYVIEW'	'OTHER OFFENSES'
'CENTRAL'	'LARCENY/THEFT'
'INGLESIDE'	'OTHER OFFENSES'
'MISSION'	'OTHER OFFENSES'
'NORTHERN'	'LARCENY/THEFT'
'PARK'	'LARCENY/THEFT'
'RICHMOND'	'LARCENY/THEFT'
'SOUTHERN'	'LARCENY/THEFT'
'TARAVAL'	'LARCENY/THEFT'
'TENDERLOIN'	'DRUG/NARCOTIC'

Athar Roshandelpoor
U46981822

Problem1 part b

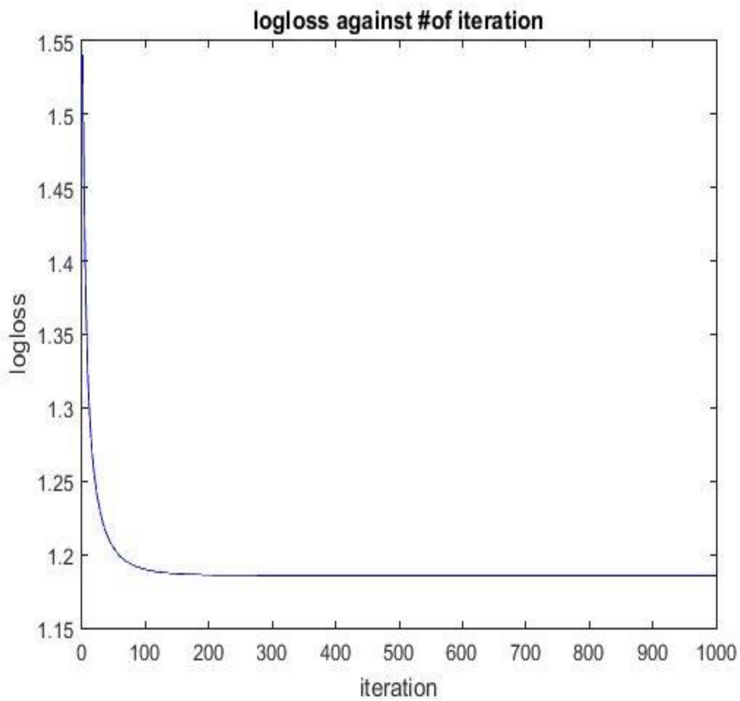


As we can see in the picture the objective function (which is **basically** Maximum Likelihood function) is decreasing in each iteration because we are using gradient descent method (we are moving in the exact opposite direction of gradient), so by each iteration we will find a better w that makes the objective function smaller.
As it is shown in the picture after 100 iteration the optimal value doesn't change that much.



Since the objective function is decreasing by each iteration, so in each iteration we are getting a better w. As a result, CCR which shows the performance of our classifier is increasing by each iteration.

Athar Roshandelpoor
U46981822



As we can see, the logloss function is decreasing by each iteration. This is because by each iteration we are getting a better value for our objective function. As a results we are having a better prediction for our test data classes. If our classifier predicts correctly the logloss function should ideally converges to zero(because the probability of each point being in its ground truth class label should be 1 and log(1) is zero)

I use log10 function in matlab for logloss (acoording to the formula in the problem set) but, I could have used ln(natural log) the difference is only a scaling factor

Athar Roshandelpoor
U46981822

Matlab code

a)

athar_matlab2_1a

```
clear
clc
load('data_SFcrime_train')
clear Address X Y


%load('data_SFcrime_test');
%making binary matrix for pdistrict, P
as place, C as Category
P=nominal(PdDistrict);
P=dummyvar(P);
C=nominal(Category);
C=dummyvar(C);
%making binary matrix for day of week
Day=weekday(Dates);
D=dummyvar(Day);

%making binary matrix for hour
Hour=hour(Dates);


Hour1=Hour+ones(length(Dates),1);
H=dummyvar(Hour1);

clear Hour1


figure;
hist(Day)
xlabel('Daynumber')
title('DayOfWeek histograms')
set(get(gca,'child'),'FaceColor','b');


figure(2);
hist(categorical(PdDistrict))
xlabel('police department district')
title('PdDistrict histograms')
set(get(gca,'child'),'FaceColor','c');
figure(3);
hist(Hour)
xlabel('hour')
title('Hour histograms')
set(get(gca,'child'),'FaceColor','g');
%it is alphabetically sorted, so the
binary crime with nth element 1 is the
%nth element in I2
I2=unique(Category);

%cat=[idx2,cat1];
%most likely hour of occurrence of each
type of crime
for i=1:length(I2)
   label=strcmp(Category,I2(i));
   Hs=sum(H(label,:));
   [~,L]=sort(Hs,'descend');
   A(i,:)=[I2(i),L(1)-1];


end
I1=unique(PdDistrict);


%most likely  type of crime within each
PdDistrict
B=[];
for i=1:length(I1)

label2=strcmp(PdDistrict(:,1),I1(i));
   Cs=sum(C(label2,:));
   [~,L]=sort(Cs,'descend');

   B=[B;I1(i),I2(L(1))];


end
load('data_SFcrime_test');
clear Address X_test Y_test



%making binary matrix for pdistrict, P
as place, C as Category
Ptest=nominal(PdDistrict_test);
Ptest=dummyvar(Ptest);

%making binary matrix for day of week
Daytest=weekday(Dates_test);
Dtest=dummyvar(Daytest);

%making binary matrix for hour
Hourtest=hour(Dates_test);
Hour1test=Hourtest+ones(length(Dates_te
st),1);
Htest=dummyvar(Hour1test);

clear Hour1test
```

Athar Roshandelpoor
U46981822

b)

athar_matlab2_1b

```
clear
clc
load('data_SFcrime_train')
clear Address X Y


%load('data_SFcrime_test');
%making binary matrix for pdistrict, P
as place, C as Category
P=nominal(PdDistrict);
P=dummyvar(P);
C=nominal(Category);
C=dummyvar(C);
%making binary matrix for day of week
Day=weekday(Dates);
D=dummyvar(Day);

%making binary matrix for hour
%Hour=hour(Dates);
load('tmp.mat','Hour');
Hour1=Hour+ones(length(Dates),1);
H=dummyvar(Hour1);

clear Hour1
%total Data
T=[H,D,P]';
%v=randperm(length(Category));
q=(1:526829);
testlabel=(526830:length(Category));
Train=T(:,q);
Test=T(:,testlabel);
I2=unique(Category);
I1=unique(PdDistrict);
%finding real class of test data
dummytest=C(testlabel,:);
[testclass,~] = find(dummytest');



%%  Step 2
% clc
% close all
% clear


%load('tmp2.mat');
dummytrain=C(q,:);
[trainclass,~] = find(dummytrain');

%gradiant descent algorithm
w=zeros(41,length(I2));
lamda=1000;
y=[];
loglossv=zeros(1,1000);
counter=0;


len_I2 = length(I2);

for i=1:1000
    disp('grad Start')
    tic
    w=w-(10^(-
5)).*gradient(w,Train,len_I2,trainclass
,lamda);
    toc
    counter=counter+1


y=[y,fNLL(w,Train,len_I2,trainclass,lam
da)];

loglossv(i)=logloss(Test,C(testlabel,:)
,w);

    %finding CCR
    %estimated class
    [~,predict]=max(w'*Test);
    labelCCR=predict==testclass';

CCR(i)=(sum(labelCCR))/length(testlabel
);

end
save('matlab_b.mat','CCR','y','loglossv
');
figure;
plot(1:length(y),y,'b')
title('objective function against #of
iteration')
xlabel('iteration')
ylabel('objective function')

figure(2);
plot(1:1000,loglossv,'b')
title('logloss against #of iteration')
xlabel('iteration')
ylabel('logloss')

figure(3);
plot(1:1000,CCR,'r')
title('CCR against #of iteration')
xlabel('iteration')
ylabel('CCR')
%savefig('CCR.jpg')
```

Functions are in next page

EC503: Learning From Data
Matlab Assignment 2.1
March 18, 2016

Athar Roshandelpoor
U46981822

fNLL

```
function
fNLL=fNLL(w,x,len_I2,trainclass,lamda)
%defining NLL
A=exp(w'*x);
A=log(sum(A));
A=sum(A);
s=0;
d=0;
for i=1:len_I2
label=trainclass==i;
    s=s+(w(:,i))'*x*(label);
    d=d+lamda*0.5*(norm(w(:,i)))^2;
end
fNLL=d-s+A;
end
```

gradient

```
%gradiant

function
Result=gradient(w,x,len_I2,trainclass,l
amda)
e=exp(x'*w);
z=sum(e,2);
Result=zeros(41,len_I2);
for i=1:len_I2


       label=trainclass==i;
       A=e(:,i)./z;
       Result(:,i)=Result(:,i)+x*(A-
label);

Result(:,i)=Result(:,i)+lamda*w(:,i);
end
end
```

logloss

```
%logloss
function logloss=logloss(Test,C,w)
s=0;
for i=1:length(Test)
    a=p(C(i,:),Test(:,i),w);
    s=s+log10(a);
end
logloss=(-1/length(Test))*s;
end
```

z

```
function z=z(w,x)
z=w'*x;
z=exp(z);
z=sum(z);

end
```

P

```
%p
function p=p(Cj,xj,w)
a=sum(Cj*w'*xj);
Z=z(w,xj);
p=exp(a)/Z;
end
```