Contents

Part f obtain maximum likelihood estimates of the parameters for the data

```
T=readtable('f7_1.csv');
[n nn] = size(T);
y= zeros(n,1);

for r = 1:n
    if T.docvis(r) > 0
        y(r,:)=1;
    else
          y(r,:)=0;
    end;
end;

[n nn]=size(y);
X =[ones(n,1), T.age, T.educ,T.hsat, T.female,T.married];
[n k]=size(X);
b=inv(X'*X)*X'*y;

% Newton ML for logit estimates
beta0 =b;
beta1=zeros(6,1);
while abs(max(beta1-beta0)) >0.000001
    %p=1./(1+exp(-X*beta0));
    p = exp(X*beta0)./(1+exp(X*beta0));
    g=sum(kron(ones(1,6),y-p).*X);
    H=-((X.*kron(ones(1,6),p))'*((1-kron(ones(1,6),p)).*X));
    beta0=beta0-inv(H)*g';
    beta1=beta0-inv(H)*g';
end;

std_l = sqrt(diag(-inv(H)));

label = [
    'Constant  ';
    'Age       ';
    'Educ      ';
    'hsat      ';
    'femle     ';
```

```
    'married    '];



disp('MLE using Newton Raphson method');
disp('Variables  b-hat-MLE    SE');
for ii = 1:size(X,2)
    fprintf('%s%10.4f%10.4f\n',label(ii,:),beta1(ii),std_l(ii));
end

%  ll for MLE
ll = sum(y.*log(p) + (1-y).*log(1-p));
fprintf('%s%10.4f\n',"log likelihood is:" ,ll);

% calculate pseudo r-squared
bzero = zeros(6,1);
pz = 1./(1+exp(-X*bzero));
llz = sum(y.*log(pz) + (1-y).*log(1-pz));

r2_pseudo = 1-ll/llz;
fprintf('%s%10.4f\n',"pseudo-r-squared  is:" ,r2_pseudo);
```

```
MLE using Newton Raphson method
Variables  b-hat-MLE    SE
Constant      1.8221    0.1076
Age           0.0124    0.0012
Educ         -0.0057    0.0058
hsat         -0.2928    0.0069
femle         0.5838    0.0272
married       0.0355    0.0317
log likelihood is:-16405.9421
pseudo-r-squared  is:    0.1338
```

part g Test hypothesis that coeffecients on female and marital status are 0.

LR Test to test b-hat-female = b-hat-married =0

```
[n nn]=size(y);
Xu =[ones(n,1), T.age, T.educ,T.hsat];
[nu ku]=size(Xu);
bu=inv(Xu'*Xu)*Xu'*y;

% Newton ML for logit estimates
betau0 =bu;
```

```
betau1=zeros(4,1);
while abs(max(betau1-betau0)) >0.000001
    %p=1./(1+exp(-X*beta0));
    pu = exp(Xu*betau0)./(1+exp(Xu*betau0));
    gu=sum(kron(ones(1,4),y-pu).*Xu);
    Hu=-((Xu.*kron(ones(1,4),pu))'*((1-kron(ones(1,4),pu)).*Xu));
    betau0=betau0-inv(Hu)*gu';
    betau1=betau0-inv(Hu)*gu';
end;

%  ll for MLE - unrestricted model
llu = sum(y.*log(pu) + (1-y).*log(1-pu));

lambda = ll-llu;
lr_stat = -2*log(lambda);
fprintf('%s%10.4f\n',"lr-stat   is:" ,lr_stat);
crit_lr = chi2inv(0.95,1);
if lr_stat > crit_lr
        disp("Reject H0");
    else
          disp("test_statistic is  -10.9078 this is way less  chi-squared 0
.95,1 = 3.8415 do not reject " + ...
              "null hypothesis. ");
end


% Wald Test for b-hat-female =0 and b-hat-married=0

R= [0,0,0,0,1,0; 0,0,0,0,0,1];
r=[0;0];
d = (R*beta0 - r);
v = R*var(beta0)*R';
w= d'*v*d;
fprintf('%s%10.4f\n',"Wald Statistic   is:" ,w);
crit = chi2inv(0.95,2);
if w > crit
        disp("Reject H0");
    else
          disp("do not reject H0");
end

% LM Test - to be done later
```

```
lr-stat   is:  -10.9078
test_statistic is  -10.9078 this is way less  chi-squared 0.95,1 = 3.8415 d
o not reject null hypothesis.
```

```
Wald Statistic   is:    0.2034
do not reject H0
```

part h test the hypothesis that all the coefficients in the model save for the constant

```
%term are equal to zero.

R1= [0,1,0,0,0,0;0,0,1,0,0,0; 0,0,0,1,0,0; 0,0,0,0,1,0; 0,0,0,0,0,1];
r1=[0;0;0;0;0];
d1 = (R1*beta0 - r1);
v1 = R1*var(beta0)*R1';
w1= d1'*v1*d1;
fprintf('%s%10.4f\n',"Wald Statistic for part h  is:" ,w1);
crit1 = chi2inv(0.95,5);
if w1 > crit1
        disp("Reject H0");
    else
          disp("do not reject H0");
end
```

```
Wald Statistic for part h  is:    0.2545
do not reject H0
```

Published with MATLAB® R2019b