optimization_machine_probit_bush106.c

```
/**/
/**/
. probit ybush black00 south hispanic00 income owner00 dwnom1n dwnom2n

Iteration 0:   log likelihood = -299.27289
Iteration 1:   log likelihood = -154.89847
Iteration 2:   log likelihood = -134.46169
Iteration 3:   log likelihood = -130.43351
Iteration 4:   log likelihood = -130.12954
Iteration 5:   log likelihood = -130.12789
Iteration 6:   log likelihood = -130.12789

Probit regression                                 Number of obs   =        432
                                                  LR chi2(7)      =     338.29
                                                  Prob > chi2     =     0.0000
Log likelihood = -130.12789                       Pseudo R2       =     0.5652

------------------------------------------------------------------------------
       ybush |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
     black00 |  -.0285973   .0097913    -2.92   0.003    -.0477878   -.0094067
       south |   .7695862   .2545783     3.02   0.003     .2706219    1.268551
  hispanic00 |  -.0089458   .0069163    -1.29   0.196    -.0225015    .0046099
      income |  -.0241489   .0126394    -1.91   0.056    -.0489217    .0006239
     owner00 |   .0235461   .0143687     1.64   0.101    -.0046161    .0517083
     dwnom1n |   2.761974   .2619813    10.54   0.000       2.2485    3.275448
     dwnom2n |   1.136417   .2339829     4.86   0.000     .6778186    1.595015
       _cons |  -.9697998   1.077738    -0.90   0.368    -3.082127    1.142528
------------------------------------------------------------------------------
```

```
double keithrules(double x[])
{
        int i, j, k1, k0, l1, l0;
        double sum=0;
        double phi=0.0;
        double xphi=0.0;
        double sumsquared=0;
        k1=0;
        k0=0;
        for(i=0;i<nrowX;i++)
        {
//              sum=0.0;
// Kludge for Intercept Term
                sum=x[1];
                for(j=0;j<ncolX;j++)
                {
                        sum=sum+X[i+j*nrowX]*x[j+2]; // Note Change Here
                }
                phi = (erf(fabs(sum)/sqrt(2.0)))/2.0 + 0.5;
                xphi = phi;
                if(sum < 0.0)phi=1.0-xphi;
                xphi = phi;
                if(xphi > 0.99999999)phi=0.9999999;
                if(xphi < 0.00000001)phi=0.0000001;
```

```
// Voted for Bush
                if(Y[i] == 1.0){
                        l1=1;
                        l0=0;
                        k1=k1+1;
                        sumsquared = sumsquared + log(phi);
                }
// Voted for Gore
                if(Y[i] != 1.0){
                        l1=0;
                        l0=1;
                        k0=k0+1;
                        sumsquared = sumsquared + log(1.0 - phi);
                }
        }
//      printf("%lf\n",-sumsquared);
        return -sumsquared;

}
```

optimization_machine_logit_bush106.c

```
/**/
/**/
. logit ybush black00 south hispanic00 income owner00 dwnom1n dwnom2n

Iteration 0:   log likelihood = -299.27289
Iteration 1:   log likelihood = -158.05142
Iteration 2:   log likelihood = -136.72201
Iteration 3:   log likelihood = -131.48217
Iteration 4:   log likelihood =  -130.9258
Iteration 5:   log likelihood =  -130.9181
Iteration 6:   log likelihood = -130.91809

Logistic regression                               Number of obs   =        432
                                                  LR chi2(7)      =     336.71
                                                  Prob > chi2     =     0.0000
Log likelihood = -130.91809                       Pseudo R2       =     0.5625

------------------------------------------------------------------------------
       ybush |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
     black00 |  -.0464407   .0172166    -2.70   0.007    -.0801846   -.0126969
       south |   1.299006   .4635389     2.80   0.005     .3904866    2.207526
  hispanic00 |  -.0151759   .0117134    -1.30   0.195    -.0381337     .007782
      income |  -.0447155   .0224499    -1.99   0.046    -.0887165   -.0007144
     owner00 |   .0451916   .0258457     1.75   0.080    -.0054651    .0958483
     dwnom1n |   4.839516   .5025589     9.63   0.000     3.854518    5.824513
     dwnom2n |    1.98889    .422375     4.71   0.000      1.16105     2.81673
       _cons |  -1.941606   1.944121    -1.00   0.318    -5.752014    1.868802
------------------------------------------------------------------------------



double keithrules(double x[])
{
        int i, j;
        double sum=0;
        double phi=0.0;
        double xphi=0.0;
        double sumsquared=0;
        k1=0;
        k0=0;
        for(i=0;i<nrowX;i++)
        {
// Kludge for Intercept Term
                sum=x[1];
                for(j=0;j<ncolX;j++)
                {
                        sum=sum+X[i+j*nrowX]*x[j+2]; // Note Change Here
                }
// Logit Setup
                phi=1.0/(1.0 + exp(-sum));
                xphi = phi;
                xphi = phi;
                if(xphi > 0.99999999)phi=0.9999999;
                if(xphi < 0.00000001)phi=0.0000001;
// Voted for Bush
                if(Y[i] == 1.0){
                        sumsquared = sumsquared + log(phi);
```

```
                }
// Voted for Gore
                if(Y[i] != 1.0){
                        sumsquared = sumsquared + log(1.0 - phi);
                }
        }
//      printf("%lf\n",-sumsquared);
        return -sumsquared;

}
```

junk_probit_2.c

```
/**/
/**/
. summ

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
     partyid |      1421    2.377903     2.06071         0          6
   incomecat |      1421    21.12386     7.76314        10         35
 incomequint |      1421    3.001407    1.415208         1          5
        race |      1421    .1491907    .3564018         0          1
         sex |      1421    .5622801    .4962807         0          1
-------------+--------------------------------------------------------
       south |      1421    .3019001    .4592438         0          1
   education |      1421    1.414497    .7154538         1          3
         age |      1421    46.00774    16.05494        20         90
       voted |      1421    1.231527    .9310102         0          3

. oprobit partyid incomequint race sex south education age

Iteration 0:   log likelihood = -2633.5601
Iteration 1:   log likelihood = -2494.5207
Iteration 2:   log likelihood = -2494.0641
Iteration 3:   log likelihood = -2494.0641

Ordered probit regression                         Number of obs   =       1421
                                                  LR chi2(6)      =     278.99
                                                  Prob > chi2     =     0.0000
Log likelihood = -2494.0641                       Pseudo R2       =     0.0530

------------------------------------------------------------------------------
     partyid |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
 incomequint |   .0141518   .0226314     0.63   0.532    -.0302049    .0585086
        race |  -1.130843   .0903177   -12.52   0.000    -1.307862   -.9538234
         sex |   .0239784    .057164     0.42   0.675     -.088061    .1360178
       south |  -.3237482   .0632066    -5.12   0.000    -.4476307   -.1998656
   education |    .232688   .0417371     5.58   0.000     .1508848    .3144911
         age |   .0032912   .0018515     1.78   0.075    -.0003377      .00692
-------------+----------------------------------------------------------------
       /cut1 |  -.5671707   .1471578                     -.8555948   -.2787466
       /cut2 |   .2302907   .1451673                      -.054232    .5148135
       /cut3 |   .4976995   .1452729                      .2129699    .7824291
       /cut4 |    .794375    .146057                      .5081085    1.080642
       /cut5 |   1.084038     .14739                      .7951585    1.372917
       /cut6 |   1.735149   .1525477                      1.436161    2.034137
------------------------------------------------------------------------------

//
// Set up for Ordered Probit -- N-chotomous Probit
//
double keithrules(double x[])
{
        int i, j, jj;
        double sum=0;
        double phi=0.0;
        double erfarg;
        double xphi=0.0;
        double sumsquared=0;
        double *phicuts;
        double *phicutsmu;
        phicuts = (double *) malloc ((nkotp+1)*sizeof(double));
        phicutsmu = (double *) malloc ((nkotp+1)*sizeof(double));
        for(i=0;i<nrowX;i++)
        {
```

```
/*  Setup For Ordinary dichotomous Probit here
 *  */
                if(nkotp==2){
/* 2-Choice Probit Uses an Intercept Term*/
                        sum=x[1];
                        for(j=0;j<ncolX;j++)
                        {
                                sum=sum+X[i+j*nrowX]*x[j+2]; // Note Change Here
                        }
                        phi = (erf(fabs(sum)/sqrt(2.0)))/2.0 + 0.5;
                        xphi = phi;
                        if(sum < 0.0)phi=1.0-xphi;
                        xphi = phi;
                        if(xphi > 0.99999999)phi=0.9999999;
                        if(xphi < 0.00000001)phi=0.0000001;
// Choice 1
                        if(Y[i] == 1.0){
                                sumsquared = sumsquared + log(phi);
                        }
// Choice 0
                        if(Y[i] != 1.0){
                                sumsquared = sumsquared + log(1.0 - phi);
                        }
                }
//
//       There is No intercept Term in ordered Probit -- the first
//        ncolX entries in x[] -- x[1] to x[ncolX] -- are the
//        coefficients on the independent variables
//
                if(nkotp > 2)
                {
                        phicutsmu[1] = -.5671707;
                        phicutsmu[2] =  .2302907;
                        phicutsmu[3] =  .4976995;
                        phicutsmu[4] =  .794375;
                        phicutsmu[5] =  1.084038;
                        phicutsmu[6] =  1.735149;
                   sum=0.0;
                   for(j=0;j<ncolX;j++)
                   {
                        sum=sum+X[i+j*nrowX]*x[j+1]; // Note Change Here

                   }
//
//  In Ordered Probit y_hat is treated as the MEAN of the normal -- the
//  cutpoints are treated like "x's"
//
                   for(jj=1;jj<nkotp;jj++)
                   {
/*  Compute Probability from -oo to (cutpoint - Y_hat)
 *  */
                        erfarg = (phicutsmu[jj]-sum);
                        phi = (erf(fabs(erfarg)/sqrt(2.0)))/2.0 + 0.5;
                        xphi = phi;
                        if(erfarg < 0.0)phi=1.0-xphi;
                        xphi = phi;
                        if(xphi > 0.99999999)phi=0.9999999;
                        if(xphi < 0.00000001)phi=0.0000001;
                        phicuts[jj]=phi;
                   }
                   phicuts[nkotp]=1.0;
                   if(Y[i] == 0)phi = phicuts[1];
```

```
                    if(1 <= Y[i] <= nkotp-1)phi = phicuts[(int)Y[i]+1]-
phicuts[(int)Y[i]];
//
                    xphi = phi;
                    if(xphi > 0.99999999)phi=0.9999999;
                    if(xphi < 0.00000001)phi=0.0000001;
                    }
                sumsquared = sumsquared + log(phi);
        }
        printf("%lf\n",-sumsquared);
        free(phicuts);
        free(phicutsmu);
        return -sumsquared;

}
```