SVM Handwriting Classification
Midterm Exam

Geoffrey Ulman
CSI747

October 2012

1 Primal Soft-Margin SVM

The primal soft-margin SVM classifier was built using the optimization problem in Equation 1.

$$\begin{aligned} \min 0.5\,(\vec{w}\cdot\vec{w}) + C\sum_{i=1}^{l}\xi_i \\ s.t. \\ \xi_i \geq 0 \\ y_i\left((\vec{x_i}\cdot\vec{w}) - b\right) \geq 1, i = 1, 2, ..., l \end{aligned} \tag{1}$$

1.1 Primal Soft-Margin AMPL Model

```
model;

# number of training examples
param l;

# number of input parameters
# (number of pixels in the handwriting digit images)
param n;

# weight on xi penalty coefficient in primal problem
param C;

# output vector (1 or -1)
param y { 1..l };

# input data
```

```
param x { 1..l, 1..n };

# hyperplane parameters
var w { 1..n };
var b;

# relaxation allowing for non-separable problems
var xi { 1..l };

minimize obj: 0.5 * ( sum { i in 1..n } w[i]^2 ) +
               C * ( sum { i in 1..l } xi[i] ) ;

s.t. nonneg { i in 1..l }: xi[i] >= 0;

s.t. hplane { i in 1..l }: y[i] *
              ( ( sum { k in 1..n }
                   x[i,k] * w[k] ) - b ) >= 1 - xi[i] ;

option solver loqo;
```

1.2 Primal Soft-Margin Results

```
LOQO 6.07: optimal solution (25 QP iterations, 25 evaluations)
primal objective 11.79795849
  dual objective 11.79795826

"option abs_boundtol 1.9472136191672977e-12;"
will change deduced dual values.

w [*] :=
 1  2.48033e-29   17  4.42975e-08   33 -1.64565e-05   49 -1.64574e-05
 2  0.91785       18  0.325352      34 -0.640034      50  0.454448
 3  0.211619      19  0.609637      35 -0.467492      51  0.383593
 4  0.0630853     20  0.281687      36 -0.11335       52 -1.16524
 5  0.124078      21  1.93283       37 -0.258944      53 -0.256678
 6  0.100893      22  0.895166      38  0.129931      54  0.332975
 7 -0.275574      23  1.27126       39 -0.976491      55  0.063559
 8  2.48033e-29   24 -0.226985      40 -0.117339      56 -0.117339
 9  2.93981e-10   25 -1.64563e-05   41 -1.64577e-05   57  0.094597
10  0.890951      26 -0.0936621     42 -1.54311       58 -0.200094
11  0.638951      27 -0.625778      43 -0.877845      59 -0.0772785
12  1.38568       28 -0.0939475     44 -0.207297      60  0.385914
13 -0.143056      29  0.795483      45 -0.0939475     61 -0.923445
14 -0.282612      30  0.586034      46 -0.771029      62  0.898336
```

```
15 -1.08685       31  0.482269      47 -0.744631      63  2.72687e-07
16  2.48033e-29   32  2.48033e-29   48 -0.0727345     64  2.48033e-29
;

xi [*] :=
  1 1.94883e-12    48 1.94899e-12    95 1.9491e-12    142 1.94896e-12
  2 1.9488e-12     49 1.96205e-12    96 1.94882e-12   143 1.95059e-12
  3 1.99438e-12    50 1.94867e-12    97 1.95209e-12   144 1.9489e-12
  4 1.94897e-12    51 1.9489e-12     98 1.96899e-12   145 1.94885e-12
  5 1.94901e-12    52 1.94901e-12    99 1.94884e-12   146 1.94896e-12
  6 1.94884e-12    53 1.95276e-12   100 2.00205e-12   147 1.95381e-12
  7 1.94839e-12    54 1.94893e-12   101 1.94853e-12   148 1.94886e-12
  8 1.94894e-12    55 1.94903e-12   102 1.94879e-12   149 1.94898e-12
  9 1.95992e-12    56 1.96944e-12   103 1.95293e-12   150 1.94891e-12
 10 1.94763e-12    57 1.94887e-12   104 1.94872e-12   151 1.94899e-12
 11 1.94896e-12    58 1.94881e-12   105 1.94773e-12   152 1.96623e-12
 12 1.94902e-12    59 1.94901e-12   106 1.94896e-12   153 1.95512e-12
 13 1.94948e-12    60 1.94884e-12   107 1.949e-12     154 1.94795e-12
 14 1.94893e-12    61 1.94887e-12   108 1.94891e-12   155 1.94882e-12
 15 1.94897e-12    62 1.94893e-12   109 1.94859e-12   156 1.94881e-12
 16 1.94884e-12    63 1.94867e-12   110 1.94862e-12   157 1.94899e-12
 17 1.94859e-12    64 1.95764e-12   111 1.94867e-12   158 1.94887e-12
 18 1.94883e-12    65 1.95231e-12   112 1.94906e-12   159 1.94889e-12
 19 1.9476e-12     66 1.94879e-12   113 1.94888e-12   160 1.94887e-12
 20 1.94895e-12    67 1.94876e-12   114 1.94889e-12   161 1.96048e-12
 21 1.94904e-12    68 1.9489e-12    115 1.94892e-12   162 1.9484e-12
 22 1.94907e-12    69 1.96043e-12   116 1.96225e-12   163 1.94899e-12
 23 1.94721e-12    70 1.94885e-12   117 1.94902e-12   164 1.94886e-12
 24 1.94796e-12    71 1.9488e-12    118 1.94889e-12   165 1.97556e-12
 25 1.94898e-12    72 1.94883e-12   119 1.94859e-12   166 1.95493e-12
 26 1.94836e-12    73 1.96299e-12   120 1.94882e-12   167 1.94874e-12
 27 1.94886e-12    74 1.95295e-12   121 1.949e-12     168 1.94874e-12
 28 1.95809e-12    75 1.9597e-12    122 1.94899e-12   169 1.95936e-12
 29 1.96172e-12    76 1.94884e-12   123 1.94874e-12   170 1.95934e-12
 30 1.94891e-12    77 1.94876e-12   124 1.94894e-12   171 1.94832e-12
 31 1.94887e-12    78 1.94834e-12   125 1.94888e-12   172 1.94903e-12
 32 1.94864e-12    79 1.94887e-12   126 1.94874e-12   173 1.9487e-12
 33 1.94882e-12    80 1.94848e-12   127 1.96742e-12   174 1.94883e-12
 34 1.94896e-12    81 1.94878e-12   128 1.94882e-12   175 1.95525e-12
 35 1.96704e-12    82 1.94891e-12   129 1.94874e-12   176 1.94873e-12
 36 1.94888e-12    83 1.94862e-12   130 1.94882e-12   177 1.94885e-12
 37 1.94899e-12    84 1.94895e-12   131 1.94984e-12   178 1.94882e-12
 38 1.95152e-12    85 1.94903e-12   132 1.94888e-12   179 1.96088e-12
 39 1.94895e-12    86 1.94879e-12   133 1.94907e-12   180 1.94749e-12
 40 1.94766e-12    87 1.94869e-12   134 1.9511e-12    181 1.9479e-12
 41 1.94889e-12    88 1.96757e-12   135 1.94902e-12   182 1.94901e-12
```

```
 42 1.94881e-12    89 1.98073e-12   136 1.94881e-12   183 1.949e-12
 43 1.94905e-12    90 1.94873e-12   137 1.94885e-12   184 1.94872e-12
 44 1.94891e-12    91 1.94904e-12   138 1.94855e-12   185 1.94782e-12
 45 1.94901e-12    92 1.94898e-12   139 1.96951e-12   186 1.94895e-12
 46 1.9489e-12     93 1.94898e-12   140 1.94885e-12
 47 1.95788e-12    94 1.94864e-12   141 1.94902e-12
;

b = 0.489299
```

Java was used to parse the AMPL results and the input data files. The hyperplane defined by $\vec{w}$ and b was then used to classify the testing data and calculate the misclassification error rate. The following Java snippet calculates the classifier output y for a set of test data (data parsing and support code omitted for brevity):

```
public static double[] calculate_y_predicted_primal(
                            List<TrainingExample> dataListTest,
                            List<TrainingExample> dataListTrain,
                            OutputGenerator out, double[] w, double b )
{
    double[] y_predicted = new double[dataListTest.size( )];

    // iterate over the training examples
    for ( int i = 0; i < dataListTest.size( ); i++ )
    {
        TrainingExample x_i = dataListTest.get( i );

        double sum = 0;
        double[] x = x_i.getInputs( );
        for ( int j = 0 ; j < x.length ; j++ )
        {
            sum += x[j] * w[j];
        }

        y_predicted[i] = sum - b;
    }

    return y_predicted;
}
```

Table 1.2 indicates that the primal soft-margin SVM classifier perfectly classified the training data set and acheived a 0.098 misclassification error rate for the testing data set for digits ”3“ and ”6“.

Table 1: Primal Soft-Margin Digit 3 vs 6 Error

Data Set	Error	95% Confidence Interval	
		Lower Bound	Upper Bound
Training	0.000	0.000	0.000
Testing	0.098	0.033	0.162

2 Dual Soft-Margin SVM

The dual soft-margin SVM classifier was built using the optimization problem in Equation 2.

$$\begin{aligned} \max \sum_{i=1}^{l} \alpha_i - 0.5 \sum_{i,j}^{l} \alpha_i \alpha_j y_i y_j \left(\vec{x_i} \cdot \vec{x_j}\right) \\ s.t. \\ 0 \geq \alpha_i \geq C, i = 1, 2, ..., l \\ \sum i = 1^l \alpha_i y_i = 0, i = 1, 2, ..., l \end{aligned} \tag{2}$$

2.1 Dual Soft-Margin AMPL Model

```
model;

# number of training examples
param l;

# number of input parameters (number of pixels in the handwriting digit images)
param n;

# weight on xi penalty coefficient in primal problem
param C;

# output vector (1 or -1)
param y { 1..l };

# input data
param x { 1..l, 1..n };

# dual problem variables and simple constraints
var a { 1..l } >= 0, <= C ;
```

```
maximize obj: ( sum { i in 1..l } a[i] ) - 0.5 * sum { i in 1..l, j in 1..l } ( a[i] * a[j]

s.t. const: sum { i in 1..l } a[i] * y[i] = 0 ;

option solver loqo;
```

2.2 Dual Soft-Margin Results

```
LOQO 6.07: optimal solution (27 QP iterations, 27 evaluations)
primal objective 11.79795836
  dual objective 11.79795849
a [*] :=
  1 1.9976e-10     48 7.20426e-11    95 5.91104e-11   142 6.79106e-11
  2 1.27001e-10    49 0.589472       96 1.98828e-10   143 0.107625
  3 2.2073         50 3.0997e-10     97 0.177339      144 1.18584e-10
  4 1.1979e-10     51 1.68386e-10    98 0.882663      145 1.64166e-10
  5 6.48215e-11    52 6.57317e-11    99 1.40616e-10   146 8.28181e-11
  6 1.59864e-10    53 0.164194      100 2.54049       147 0.251072
  7 3.63985e-10    54 1.2575e-10    101 2.97498e-10   148 1.45562e-10
  8 7.66275e-11    55 7.49007e-11   102 1.95441e-10   149 9.70925e-11
  9 0.535794       56 0.917851      103 0.157023      150 1.76263e-10
 10 9.23772e-09    57 1.24877e-10   104 3.33413e-10   151 6.39504e-11
 11 9.36826e-11    58 2.31782e-10   105 0.0688523     152 0.800829
 12 6.56015e-11    59 5.43169e-11   106 8.39285e-11   153 0.32975
 13 0.0446074      60 1.15747e-10   107 6.6022e-11    154 2.20037e-09
 14 8.29934e-11    61 8.20516e-11   108 9.85636e-11   155 2.87449e-10
 15 6.93563e-11    62 1.18503e-10   109 3.30952e-10   156 1.96334e-10
 16 1.72574e-10    63 3.23511e-10   110 2.55377e-10   157 6.93557e-11
 17 8.41272e-10    64 0.478371      111 2.75346e-10   158 1.71022e-10
 18 1.90058e-10    65 0.207781      112 5.29101e-11   159 1.49685e-10
 19 1.61131e-08    66 1.36269e-10   113 1.99511e-10   160 1.19925e-10
 20 1.01319e-10    67 2.79201e-10   114 8.34089e-11   161 0.455275
 21 6.07639e-11    68 9.00848e-11   115 9.80581e-11   162 7.32281e-10
 22 5.67325e-11    69 0.598027      116 0.744559      163 7.09092e-11
 23 2.30412e-08    70 1.17194e-10   117 6.36505e-11   164 1.52059e-10
 24 3.88004e-09    71 1.66061e-10   118 8.21563e-11   165 1.04669
 25 7.83271e-11    72 1.39506e-10   119 4.60396e-10   166 0.296604
 26 0.0833737      73 0.650229      120 1.98776e-10   167 3.39772e-10
 27 1.41541e-10    74 0.354643      121 7.78976e-11   168 2.22992e-10
 28 0.25833        75 0.389362      122 6.92345e-11   169 0.608202
 29 0.545924       76 1.50777e-10   123 2.64229e-10   170 0.407781
 30 1.21571e-10    77 2.26633e-10   124 1.10743e-10   171 2.84462e-09
 31 1.10958e-10    78 6.2839e-10    125 2.1568e-10    172 8.71846e-11
```

```
 32 4.02429e-10     79 9.16963e-11    126 5.19841e-10    173 4.46608e-10
 33 1.58059e-10     80 6.70356e-10    127 0.950716       174 1.59352e-10
 34 6.54409e-11     81 2.88238e-10    128 2.07806e-10    175 0.22699
 35 0.84129         82 7.92356e-11    129 2.78683e-10    176 3.56063e-10
 36 1.29483e-10     83 3.26451e-10    130 1.93225e-10    177 1.54245e-10
 37 8.2379e-11      84 9.693e-11      131 0.0586039      178 1.72984e-10
 38 0.0945982       85 5.95996e-11    132 1.26735e-10    179 0.53314
 39 8.40886e-11     86 2.59713e-10    133 5.67888e-11    180 2.6983e-09
 40 8.31644e-08     87 4.7611e-10     134 0.117343       181 1.8018e-09
 41 2.90003e-10     88 0.839255       135 5.4567e-11     182 4.83902e-11
 42 1.8237e-10      89 1.56632        136 1.95172e-10    183 7.09831e-11
 43 4.82581e-11     90 3.49813e-10    137 1.75019e-10    184 3.84511e-10
 44 1.03927e-10     91 4.99832e-11    138 9.10471e-10    185 6.26339e-06
 45 7.33129e-11     92 7.82871e-11    139 1.0364         186 1.06007e-10
 46 1.13071e-10     93 7.70651e-11    140 2.21336e-10
 47 0.43124         94 6.9625e-10     141 4.93902e-11
;
```

The value of b was calculated for all support vectors (those with $0 < \alpha_i < C$) as a check on the correctness of the solution. The table below displays the calculated b values for each such α. The final b value used in the classification of the testing data was the average of these b values.

```
#alpha index, alpha value, calculated b
2 2.2073 0.489359470530
8 0.5358 0.489342244707
12 0.0446 0.489352229235
25 0.0834 0.489321874750
27 0.2583 0.489340402111
28 0.5459 0.489313208511
34 0.8413 0.489354952717
37 0.0946 0.489357078408
46 0.4312 0.489370838373
48 0.5895 0.489351846873
52 0.1642 0.489345982647
55 0.9179 0.489340291918
63 0.4784 0.489350434978
64 0.2078 0.489360781074
68 0.5980 0.489359352524
72 0.6502 0.489364796970
73 0.3546 0.489354535040
74 0.3894 0.489345701934
87 0.8393 0.489357037101
88 1.5663 0.489378658051
```

Table 2: Dual Soft-Margin Digit 3 vs 6 Error

Data Set	Error	95% Confidence Interval	
		Lower Bound	Upper Bound
Training	0.000	0.000	0.000
Testing	0.098	0.033	0.162

```
96 0.1773 0.489347052712
97 0.8827 0.489356109730
99 2.5405 0.489380927888
102 0.1570 0.489336299179
104 0.0689 0.489342778264
115 0.7446 0.489338976209
126 0.9507 0.489354731848
130 0.0586 0.489331827843
133 0.1173 0.489363777234
138 1.0364 0.489358857171
142 0.1076 0.489355972548
146 0.2511 0.489363515355
151 0.8008 0.489358797912
152 0.3298 0.489354717593
160 0.4553 0.489348269770
164 1.0467 0.489327661539
165 0.2966 0.489326174068
168 0.6082 0.489343398834
169 0.4078 0.489342179544
174 0.2270 0.489345274282
178 0.5331 0.489354172543
```

Table 2.2 indicates that the dual soft-margin SVM classifier perfectly classified the training data set and acheived a 0.098 misclassification error rate for the testing data set for digits ”3“ and ”6“. This is identical to the results achieved for the primal problem (which makes sense because the formulations should be equivalent).

3 Dual Polynomial SVM

The dual polynomial SVM classifier was built using the optimization problem in Equation 3.

$$
\begin{aligned}
\max \sum_{i=1}^{l} \alpha_i - 0.5 \sum_{i,j}^{l} \alpha_i \alpha_j y_i y_j \left(\alpha \left(\vec{x_i} \cdot \vec{x_j}\right) + \beta\right)^d \\
s.t. \\
0 \geq \alpha_i \geq C, i = 1, 2, ..., l \\
\sum i = 1^l \alpha_i y_i = 0, i = 1, 2, ..., l
\end{aligned}
\tag{3}
$$

3.1 Dual Polynomial AMPL Model

```
model;

# number of training examples
param l;

# number of input parameters (number of pixels in the handwriting digit images)
param n;

# weight on xi penalty coefficient in primal problem
param C;

# parameters for polynomial machine kernel
param alpha;
param beta;
param delta;

# output vector (1 or -1)
param y { 1..l };

# input data
param x { 1..l, 1..n };

# dual problem variables and simple constraints
var a {1..l} >= 0, <= C;

maximize obj: sum { i in 1..l } a[i] - 0.5 * sum { i in 1..l, j in 1..l } a[i] * a[j] * y[i]

s.t. const: sum { i in 1..l } a[i] * y[i] = 0;

option solver loqo;
```

3.2 Dual Polynomial Results

```
LOQO 6.07: optimal solution (22 QP iterations, 22 evaluations)
primal objective 2867.882418
  dual objective 2867.882425
a [*] :=
  1   1.02531e-07    48   8.61577e-09    95   2.10839e-08   142   1.48999e-08
  2  98.4474         49   4.61188e-07    96  34.9011        143  27.6442
  3 100              50  30.4183         97  37.7857        144   4.66503
  4   6.5328e-08     51   1.86643e-06    98  10.8704        145   2.7213e-08
  5   1.2032e-07     52   1.11213e-07    99  25.9701        146   1.4305e-08
  6  12.9816         53  97.8993        100 100             147  25.5623
  7  28.3116         54   4.77467e-08   101  23.6605        148   9.86569e-09
  8   6.33472e-08    55   7.64781e-09   102   2.27673e-08   149   1.32595e-08
  9  44.7894         56   8.51246       103   5.24267e-07   150   2.33402e-08
 10  26.3581         57   1.39012e-08   104  10.3686        151   1.542e-08
 11  36.7381         58   4.29743e-07   105  39.3451        152   2.82917
 12   1.07092e-07    59   2.05962e-08   106   4.16072e-07   153 100
 13  74.2736         60 100             107   2.71187e-08   154 100
 14  11.9111         61   8.48449e-08   108   4.24606       155  39.1819
 15   1.82933e-08    62   3.15818e-08   109 100             156   8.86615
 16   1.35129e-07    63  16.6566        110  36.8726        157   3.07667e-08
 17  41.4348         64  63.5389        111  53.4906        158   1.05893e-07
 18  20.8818         65  36.6983        112   7.23013       159   2.16737e-05
 19   4.30859e-06    66 100             113  19.614         160   3.15004e-08
 20   3.23797        67   0.000156624   114  10.0767        161  67.9115
 21   1.43615e-08    68  89.4563        115 100             162   1.16297e-07
 22 100              69 100             116 100             163   8.37258e-09
 23 100              70   2.0411e-08    117   8.08354e-09   164   1.87303e-08
 24 100              71   1.68419e-08   118 100             165  22.6818
 25   1.75544e-08    72  36.2032        119 100             166  96.2878
 26 100              73  52.5948        120  67.3897        167   1.95622e-07
 27 100              74  59.7171        121 100             168  57.8253
 28  40.8574         75   2.47863       122   2.3803e-08    169  17.8739
 29 100              76  15.1736        123  25.873         170   6.14482
 30   2.16068e-07    77   3.22647e-06   124   7.29567e-08   171   1.25933
 31   2.03066e-06    78  49.6606        125   4.94565e-08   172   1.04854e-07
 32   3.85756e-08    79   8.19333       126  48.5045        173   9.48377e-08
 33  59.5537         80   1.37222       127 100             174  18.5993
 34   1.50233e-08    81   4.01854e-08   128   1.81162e-07   175  37.0473
 35   1.54028e-07    82   4.6698e-08    129  37.4749        176 100
 36   1.62842e-08    83  40.1234        130   5.07964e-08   177   2.07399e-08
 37   1.49206e-08    84   3.85788e-08   131  34.2697        178   3.55846e-08
 38  16.7662         85  51.4942        132   7.80473e-08   179 100
 39   1.98667e-08    86  17.3654        133   6.14671e-09   180 100
```

```
 40   2.53905e-07    87  81.9708        134   2.76429e-08   181  73.9895
 41   0.103716       88 100             135   1.61786e-08   182   4.9928e-08
 42   5.10616e-08    89 100             136   2.2634e-07    183   1.38181e-08
 43   3.01842e-08    90   4.56927e-08   137   1.14858e-08   184   5.57055e-07
 44   1.19128e-08    91   1.29279e-08   138   2.34395e-08   185  25.356
 45   1.11601e-08    92   9.98219       139  82.4303        186   1.12149e-08
 46   2.1805e-08     93  12.6638        140   1.308e-08
 47  13.8354         94  68.5562        141   1.34735e-08
;
```

The value of b was calculated for all support vectors (those with $0 < \alpha_i < C$) in the same manner as for the dual soft-margin problem in Section 2.

```
#alpha value, calculated b
98.4474 0.003688057498
12.9816 0.003688219777
28.3116 0.003689360280
44.7894 0.003688990222
26.3581 0.003688019794
36.7381 0.003689233929
74.2736 0.003688981995
11.9111 0.003688561909
41.4348 0.003689095055
20.8818 0.003689023799
3.2380 0.003688472784
40.8574 0.003688865371
59.5537 0.003688906990
16.7662 0.003688761357
0.1037 0.003695737907
13.8354 0.003688425517
30.4183 0.003688270055
97.8993 0.003688426899
8.5125 0.003688976897
16.6566 0.003689144953
63.5389 0.003689011298
36.6983 0.003688580128
89.4563 0.003688441483
36.2032 0.003688147806
52.5948 0.003688233328
59.7171 0.003689335758
2.4786 0.003688938989
15.1736 0.003689095760
49.6606 0.003688668794
8.1933 0.003688627345
```

1.3722 0.003688150932
40.1234 0.003689538042
51.4942 0.003688334140
17.3654 0.003688354074
81.9708 0.003688425479
9.9822 0.003688209154
12.6638 0.003687213091
68.5562 0.003688553410
34.9011 0.003690060290
37.7857 0.003690222677
10.8704 0.003691053320
25.9701 0.003689453009
23.6605 0.003688199671
10.3686 0.003689730633
39.3451 0.003689168432
4.2461 0.003689282127
36.8726 0.003688797655
53.4906 0.003688706942
7.2301 0.003688278618
19.6140 0.003687876768
10.0767 0.003687977832
67.3897 0.003688923886
25.8730 0.003688569845
48.5045 0.003689123141
37.4749 0.003688113516
34.2697 0.003689853627
82.4303 0.003688933590
27.6442 0.003690255237
4.6650 0.003689246088
25.5623 0.003689210145
2.8292 0.003689643357
39.1819 0.003689726944
8.8662 0.003689347001
67.9115 0.003688349042
22.6818 0.003688557541
96.2878 0.003688524804
57.8253 0.003689285044
17.8739 0.003688683644
6.1448 0.003689311403
1.2593 0.003688751264
18.5993 0.003689598874
37.0473 0.003688764178
73.9895 0.003688189170
25.3560 0.003688246100

Table 3: Dual Polynomial Digit 3 vs 6 Error

Data Set	Error	95% Confidence Interval	
		Lower Bound	Upper Bound
Training	0.000	0.000	0.000
Testing	0.037	-0.004	0.077

Table 3.2 indicates that the dual polynomial SVM classifier perfectly classified the training data set and acheived a 0.037 misclassification error rate for the testing data set for digits ”3“ and ”6“.

4 Dual Radial SVM

The dual radial SVM classifier was built using the optimization problem in Equation 4.

$$\begin{aligned} \max \sum_{i=1}^{l} \alpha_i - 0.5 \sum_{i,j}^{l} \alpha_i \alpha_j y_i y_j e^{-\gamma \|x - x_i\|^2} \\ s.t. \\ 0 \geq \alpha_i \geq C, i = 1, 2, ..., l \\ \sum i = 1^l \alpha_i y_i = 0, i = 1, 2, ..., l \end{aligned} \tag{4}$$

4.1 Dual Radial AMPL Model

```
model;

# number of training examples
param l;

# number of input parameters (number of pixels in the handwriting digit images)
param n;

# weight on xi penalty coefficient in primal problem
param C;

# parameters for radial basis function kernel
param gamma;

# output vector (1 or -1)
```

```
param y { 1..l };

# input data
param x { 1..l, 1..n };

# dual problem variables and simple constraints
var a {1..l} >= 0, <= C;

maximize obj: sum { i in 1..l } a[i] - 0.5 * sum { i in 1..l, j in 1..l } ( a[i] * a[j] * y

s.t. const: sum { i in 1..l } a[i] * y[i] = 0;

option solver loqo;
```

4.2 Dual Radial Results

```
LOQO 6.07: optimal solution (26 QP iterations, 26 evaluations)
primal objective 60.01665461
  dual objective 60.01665488
a [*] :=
  1 3.16707e-08    48 1.70355e-09    95 5.49085e-10   142 1.05146e-09
  2 1.18067        49 0.868939       96 0.864484      143 1.21452
  3 5.58637        50 1.04674        97 1.69305       144 0.206086
  4 1.03344e-09    51 0.0535025      98 0.930714      145 1.01719e-09
  5 1.26604e-09    52 7.75998e-10    99 0.967807      146 7.6497e-10
  6 0.798701       53 1.81665       100 5.45732       147 1.67891
  7 0.779093       54 1.4995e-09    101 1.66102       148 3.67459e-09
  8 2.31206e-09    55 7.02855e-10   102 2.65541e-09   149 8.94347e-10
  9 2.16352        56 1.25845       103 0.733869      150 2.06099e-09
 10 0.553355       57 7.93768e-10   104 0.268584      151 2.48027e-09
 11 0.24533        58 0.199458      105 0.60435       152 1.51394
 12 1.26222e-09    59 8.87674e-10   106 1.69988e-09   153 3.63688
 13 1.55436        60 1.191         107 8.83217e-10   154 2.89767
 14 5.96262e-09    61 8.97172e-09   108 9.48199e-09   155 2.80291e-09
 15 7.91864e-10    62 8.80557e-10   109 1.07512e-08   156 2.41843e-06
 16 2.39761e-08    63 0.0286478     110 1.16225       157 7.2857e-10
 17 1.50561        64 1.69873       111 1.26532       158 1.65738e-09
 18 0.227814       65 2.97351e-07   112 1.6898e-09    159 1.04711e-07
 19 5.62155e-09    66 1.96522       113 0.0303951     160 6.60597e-09
 20 2.00563e-09    67 6.01402e-09   114 0.169084      161 0.748281
 21 6.54948e-10    68 2.76958e-09   115 1.14146e-09   162 5.06856e-09
 22 4.87895e-10    69 4.10865       116 2.46          163 8.29155e-10
 23 0.931043       70 1.82056e-09   117 6.56501e-10   164 0.0557784
 24 0.905486       71 0.141962      118 9.49124e-10   165 0.808062
```

Table 4: Dual Radial Digit 3 vs 6 Error

Data Set	Error	95% Confidence Interval	
		Lower Bound	Upper Bound
Training	0.000	0.000	0.000
Testing	0.037	-0.004	0.077

```
 25 5.37478e-10     72 0.614427      119 1.83812       166 0.319832
 26 3.617           73 3.43688       120 6.58821e-09   167 0.67577
 27 1.63219e-08     74 2.79481       121 1.07336e-09   168 2.02057
 28 1.45319e-08     75 0.322135      122 4.62182e-10   169 1.19308
 29 0.97809         76 6.00011e-07   123 2.02153e-09   170 0.436734
 30 1.67667e-09     77 0.72499       124 2.11181e-09   171 2.10647
 31 2.41782e-09     78 3.88469       125 1.6707e-09    172 0.575735
 32 8.35868e-07     79 1.14933       126 9.42607e-09   173 0.097306
 33 9.17644e-07     80 0.24544       127 5.05721       174 0.197485
 34 1.16409e-09     81 2.87338e-09   128 4.00419e-08   175 0.352128
 35 1.16857         82 1.59776e-09   129 2.3046e-09    176 0.135637
 36 7.94503e-10     83 1.72584       130 2.90361e-09   177 1.60422e-09
 37 7.79545e-10     84 2.12031e-09   131 1.30882       178 6.29554e-09
 38 0.807104        85 8.67285e-10   132 2.46798e-09   179 2.02892
 39 2.58067e-09     86 2.80611e-09   133 1.01686e-09   180 3.55015
 40 0.84768         87 3.60788e-08   134 0.815431      181 1.00169e-08
 41 0.150547        88 2.10957       135 6.25167e-10   182 5.82264e-10
 42 2.58486e-07     89 3.02445       136 0.511569      183 8.27788e-10
 43 7.28255e-10     90 2.02706e-09   137 1.09742e-09   184 0.490051
 44 1.5882e-09      91 5.1347e-10    138 1.09432e-08   185 1.26176
 45 7.3457e-10      92 4.73949e-09   139 2.71389       186 1.02046e-09
 46 1.21012e-09     93 0.244813      140 1.44809e-09
 47 1.36097         94 1.30161       141 1.02309e-09
;
```

Table 4.2 indicates that the dual radial SVM classifier perfectly classified the training data set and achieved a 0.037 misclassification error rate for the testing data set for digits "3" and "6". This means that the radial and polynomial kernels actually performed identically well (but better than the dot product kernel machine). The polynomial kernel was chosen for the full problem.

5 All Digits Polynomial Kernel

Because of the size of the full classification problem, the ten hyperplanes (classifying each digit versus all others) were calculated using the NEOS server. The following is an example output from AMPL for the model defining the hyperplane separating digit ”9“ from other digits.

```
*************************************************************

   NEOS Server Version 5.0
   Job#     : 322513
   Password : ZDMlRVXE
   Solver   : nco:LOQO:AMPL
   Start    : 2012-10-13 15:37:19
   End      : 2012-10-13 15:38:32
   Host     : neos-4.chtc.wisc.edu

   Disclaimer:

   This information is provided without any express or
   implied warranty. In particular, there is no warranty
   of any kind concerning the fitness of this
   information  for any particular purpose.
*************************************************************
Job 322513 sent to neos-4.chtc.wisc.edu
password: ZDMlRVXE
---------- Begin Solver Output -----------
Executing /opt/neos/Drivers/loqo-ampl/loqo-driver.py at time: 2012-10-13 20:40:06.404104
File exists
You are using the solver loqo.

%% YOUR COMMENTS %%%%%%%%%%%%%%%%%%%%%%%
Digit 9
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Executing AMPL.
processing data.
processing commands.

930 variables, all nonlinear
1 constraint, all linear; 930 nonzeros
1 equality constraint
1 nonlinear objective; 930 nonzeros.

LOQO 6.07: optimal solution (38 QP iterations, 112 evaluations)
primal objective 11434.38476
```

```
  dual objective 11434.3848
a [*] :=
  1   3.85436e-08   234   1.05483e-08   467   2.41951e-08   700  18.4227
  2   2.33818e-08   235   3.78196e-08   468  19.8573        701   4.7727e-08
  3   2.91404e-08   236   2.26497e-08   469   3.67114e-08   702   1.25251e-07
  4   5.71453       237   1.5172        470   6.27438e-07   703   3.03899e-08
  5   1.38018e-08   238   3.43765e-08   471   2.72169e-08   704 100
  6   7.05322       239   2.63027e-08   472   6.32506e-08   705   4.71152e-08
  7   2.3794e-08    240   1.57388e-08   473   3.68133e-08   706   1.5501e-07
  8   4.18764e-08   241   1.36029e-08   474   0.353602      707   2.60628e-08
  9   2.11997e-06   242   7.46301e-09   475   6.93911e-07   708   4.67655e-08
 10   5.60904e-08   243   5.82155e-09   476   1.47011e-08   709   4.23368e-08
 11   7.94764e-08   244   3.72253e-08   477   3.08854e-08   710 100
 12   3.58911e-08   245   9.042e-07     478  17.6082        711   2.41813e-07
 13   2.96835e-08   246   6.70782e-08   479   1.77021e-08   712  75.9297
 14   1.61716e-08   247   1.04641e-07   480   7.07884e-08   713   4.09782e-07
 15   1.47334e-07   248   2.6709e-08    481   6.68e-08      714   3.00134e-07
 16  29.3529        249   2.59011e-08   482   6.71728e-08   715   2.32582e-07
 17   9.64202e-08   250   7.87914e-08   483   3.86652e-08   716  18.5722
 18   3.38414e-08   251   8.84796e-08   484   6.63917e-08   717   6.01327e-08
 19   1.95048e-07   252   2.0955e-08    485   3.57116       718   2.26718e-07
 20   5.72279       253   1.57393e-08   486   8.43447e-08   719   2.79273e-08
 21   1.90208e-08   254   1.94778e-08   487   1.25587e-08   720   1.85065e-08
 22   2.05563e-08   255   2.05139e-08   488   5.93334e-08   721   7.0089
 23   2.22771e-07   256   5.84025e-09   489  36.5089        722   1.12219e-08
 24   6.59432e-08   257   8.92567e-09   490   2.439e-08     723   5.0506e-08
 25  87.9023        258   1.17714e-08   491   6.27438e-07   724   1.69498e-07
 26  54.8041        259   9.49884e-09   492   5.93334e-08   725 100
 27   1.39902e-06   260   9.14694e-09   493   3.72785e-08   726  13.7361
 28   0.000112796   261   2.53348e-08   494   9.11471e-08   727   4.73843e-08
 29   3.49784e-08   262   5.68235e-06   495   1.351e-07     728   8.65427
 30   3.49784e-08   263   3.99734e-08   496   3.1964e-08    729   2.41812e-08
 31   1.91778e-08   264   1.27274e-08   497   0.517183      730  51.0242
 32   5.77159e-08   265   1.44707e-08   498   3.4741e-08    731   8.55205e-08
 33   6.71962e-08   266   1.99829e-08   499   2.01095e-08   732   5.28158e-08
 34   4.73067e-08   267   3.90531e-08   500   4.46076e-08   733   3.55383e-08
 35   1.42603e-08   268   1.41763e-07   501   1.37996e-07   734   1.37807e-06
 36   1.7283e-08    269   6.40695e-08   502   7.66502e-08   735  40.0125
 37   2.86341e-08   270   2.03313e-08   503   9.57023e-09   736   3.15026e-08
 38   3.35671e-08   271   1.36457e-06   504   1.22933e-08   737   6.20972e-08
 39   6.37693e-08   272   2.20462e-08   505   9.57944e-09   738   3.01406e-08
 40   8.31286e-08   273   2.03325e-08   506   8.39667e-09   739  12.0501
 41   3.12044e-08   274   2.16373e-08   507   3.35393e-08   740   9.35406e-08
 42   3.12408e-08   275   6.13271e-08   508   5.49397e-08   741   6.73344e-08
 43   1.59441e-08   276   1.20099e-08   509  15.1377        742   2.2836e-07
 44   4.5707e-08    277  28.6054        510   2.33238e-08   743   6.32906e-08
```

45	2.4491e-08	278	1.45522e-08	511	1.49538e-08	744	24.7776
46	3.98829e-08	279	1.30016e-08	512	1.47931e-08	745	4.46246e-08
47	3.0174e-08	280	42.4313	513	8.93877e-09	746	1.59093e-08
48	2.16635e-08	281	2.49875e-08	514	2.06621e-08	747	1.87348e-08
49	2.99249e-08	282	3.58647e-08	515	3.76778e-08	748	8.29901e-09
50	4.79835e-08	283	31.6708	516	9.17331e-08	749	9.35097e-09
51	3.10036e-08	284	1.27302e-08	517	2.03913e-08	750	6.06604e-08
52	4.69671e-08	285	1.63198e-08	518	6.81754e-08	751	4.45899e-08
53	31.0034	286	4.8192e-08	519	6.72317e-07	752	2.68031e-08
54	1.65889e-07	287	1.47451e-08	520	3.23831e-08	753	63.8287
55	1.42681e-08	288	2.21206e-08	521	12.4197	754	5.39902e-08
56	2.96805e-08	289	10.9079	522	5.6307e-08	755	1.65101e-08
57	2.59437e-08	290	6.1962e-08	523	2.13867e-08	756	1.7601e-08
58	2.8481e-08	291	4.4776e-08	524	1.09866e-07	757	6.68351e-08
59	1.32139e-08	292	3.91747e-08	525	5.44633e-08	758	2.26889e-07
60	3.67646e-08	293	2.34207e-08	526	1.2047e-07	759	5.46958e-08
61	10.4148	294	4.33321e-08	527	1.69765e-08	760	4.07936e-08
62	3.26254e-08	295	2.37069e-08	528	37.345	761	2.70216e-08
63	8.23382e-08	296	2.44745e-08	529	5.21229	762	2.53618e-08
64	5.23811e-08	297	2.47458e-08	530	29.8444	763	30.3426
65	1.34593e-07	298	4.81607e-08	531	2.354e-08	764	2.88914e-08
66	3.82011e-08	299	5.47056e-08	532	1.40526e-08	765	3.68228e-08
67	5.96743e-08	300	2.32321e-08	533	4.2098e-08	766	3.29692e-08
68	3.26868e-08	301	1.42289e-07	534	70.0837	767	7.22012e-08
69	2.96036e-08	302	3.37744e-08	535	9.13876e-09	768	98.6327
70	3.44339e-08	303	4.36211e-08	536	8.75874e-09	769	5.68451e-08
71	2.76635e-08	304	2.60821e-08	537	2.73822e-07	770	27.4658
72	4.66116e-08	305	2.7947e-08	538	1.74901e-06	771	5.39902e-08
73	3.88695e-08	306	0.175107	539	72.0049	772	1.91334e-07
74	5.16296e-08	307	2.03907e-07	540	2.56229e-08	773	5.58844e-08
75	1.05899e-07	308	9.98471	541	5.09203e-08	774	2.94872e-08
76	2.16298e-08	309	5.19411e-08	542	2.37141e-07	775	39.8487
77	3.82416e-08	310	2.503e-08	543	1.33879e-08	776	80.8929
78	1.76589e-08	311	1.87832e-08	544	2.4519e-06	777	1.59144e-08
79	1.22373e-07	312	4.95507e-07	545	2.31918e-08	778	1.31574e-07
80	4.0689e-06	313	28.4913	546	5.18578	779	100
81	9.3058e-08	314	1.34279e-08	547	3.33864e-08	780	1.63384e-08
82	4.28731e-08	315	20.7138	548	9.4663e-08	781	9.08261e-08
83	7.14995e-08	316	1.42517e-08	549	2.57318e-08	782	4.12276e-08
84	6.92773e-08	317	55.0932	550	2.735e-08	783	1.72565e-08
85	2.54506e-08	318	7.24769e-09	551	2.10952e-08	784	1.28544e-07
86	1.23417e-07	319	6.07053e-09	552	100	785	28.4671
87	3.87228e-08	320	1.44993e-08	553	14.6866	786	3.5382e-08
88	5.19699e-07	321	1.14318e-08	554	51.7308	787	1.11575e-08
89	9.56465e-08	322	9.4297e-08	555	20.6688	788	1.1578e-08
90	2.26018e-08	323	2.15618e-08	556	4.88641	789	9.90283e-09

91	4.57081e-08	324	3.67134	557	8.95684e-08	790	13.6659
92	3.4908e-08	325	1.55879e-08	558	7.12494e-08	791	6.051e-08
93	1.31665	326	4.23239e-08	559	1.73796e-08	792	1.25531e-08
94	1.84953e-07	327	1.12298e-08	560	3.04043e-08	793	8.28471e-09
95	1.04024e-06	328	1.35997e-08	561	1.73769e-08	794	1.5172
96	1.12866e-07	329	1.47122e-08	562	1.66743e-08	795	1.05734e-07
97	100	330	0.637703	563	1.08723e-08	796	5.1897e-08
98	100	331	8.05242e-08	564	19.1401	797	1.27939e-07
99	2.58436e-08	332	4.10269e-08	565	1.21165e-08	798	11.4375
100	6.19801e-08	333	4.22526e-08	566	6.72943e-09	799	2.77049e-08
101	5.04257e-08	334	2.51607e-06	567	3.20842e-08	800	2.38846e-08
102	1.11166e-07	335	7.09107e-09	568	1.71234e-08	801	1.34923e-08
103	4.89188e-08	336	1.54768e-08	569	3.9541e-08	802	43.5287
104	4.62787e-07	337	9.31113e-09	570	1.73492e-08	803	4.02628e-08
105	1.28079e-07	338	26.5263	571	1.95366e-08	804	1.95869e-07
106	32.7716	339	8.15323e-08	572	1.2651e-08	805	33.8906
107	8.49446e-08	340	3.67562e-08	573	2.58757e-08	806	5.76479e-08
108	7.58618e-08	341	76.5891	574	5.1005e-08	807	2.80831e-08
109	1.83277e-07	342	2.86211e-08	575	75.4697	808	5.47591e-08
110	1.28079e-07	343	5.01951e-08	576	4.62622e-08	809	5.33724e-08
111	1.84953e-07	344	1.7571e-06	577	3.77306e-08	810	2.58255e-08
112	1.4635e-07	345	2.66057e-08	578	2.72308e-08	811	6.94816
113	1.37781e-07	346	2.23407e-08	579	2.23639e-08	812	1.10208e-08
114	1.37781e-07	347	100	580	2.14761e-07	813	32.9226
115	8.79441e-06	348	50.2192	581	3.03802e-08	814	24.7422
116	3.71935e-06	349	2.44918e-08	582	1.12664e-08	815	38.3727
117	11.9145	350	1.06024e-08	583	3.63946e-08	816	88.5056
118	3.38051e-07	351	1.2334e-08	584	8.5299e-08	817	18.4125
119	3.38051e-07	352	1.77221e-08	585	1.99742e-08	818	1.84441e-08
120	3.87516e-07	353	1.21483e-08	586	4.24965e-08	819	2.72733e-08
121	1.84953e-07	354	1.26128e-08	587	1.81451e-08	820	1.69932e-08
122	1.84953e-07	355	3.02331e-08	588	1.8592e-08	821	1.03389e-08
123	3.87516e-07	356	5.59393e-08	589	1.65819e-08	822	1.22589e-08
124	1.28079e-07	357	1.35925e-08	590	1.91717e-08	823	9.70715e-07
125	47.6849	358	1.43543e-08	591	4.3219e-08	824	6.40877e-08
126	2.80476e-07	359	3.23171e-08	592	1.76876e-08	825	1.1659e-08
127	1.3426e-07	360	3.06975	593	4.00608e-08	826	1.97428e-08
128	1.64991e-06	361	18.6384	594	8.40152e-08	827	64.2707
129	1.74233e-08	362	6.1922	595	2.79754e-08	828	10.215
130	2.07108e-08	363	78.9539	596	3.14686e-08	829	1.33594e-08
131	4.8475e-08	364	6.41814e-08	597	4.74008e-08	830	2.33468e-08
132	2.3094e-07	365	1.75939e-07	598	7.67663e-09	831	2.88313e-08
133	1.28079e-07	366	100	599	7.0626e-09	832	1.28804e-07
134	9.09461e-08	367	1.04522e-07	600	1.45123e-08	833	2.3411e-07
135	5.04488e-08	368	3.25504e-06	601	4.9579e-08	834	6.58494e-09
136	5.65006e-08	369	2.13498e-08	602	2.21507e-08	835	8.49227e-09

137	54.0187	370	2.90897e-08	603	1.6761e-08	836	45.5049
138	1.04783e-07	371	2.03738e-08	604	2.26405e-08	837	1.58665e-07
139	4.32154e-08	372	4.57929e-08	605	1.75805e-08	838	100
140	3.27177e-08	373	57.4036	606	7.66174e-09	839	100
141	19.8018	374	80.5618	607	8.65486e-09	840	75.5841
142	1.65822e-08	375	3.5791e-08	608	2.87527e-08	841	100
143	1.28079e-07	376	5.8046e-07	609	8.11502e-08	842	1.75801
144	1.36136e-07	377	7.80957	610	1.57627e-08	843	68.7307
145	4.42464	378	100	611	2.81648e-08	844	100
146	4.42464	379	2.3624e-08	612	1.21696e-08	845	100
147	2.25185	380	21.3358	613	9.14794e-09	846	96.2966
148	2.12267e-08	381	3.34692	614	1.50895e-08	847	100
149	1.97217e-06	382	2.43224e-08	615	6.377e-08	848	100
150	9.48228e-08	383	19.7623	616	1.30377e-08	849	100
151	6.07154e-08	384	1.7773e-08	617	7.55313e-09	850	100
152	3.28686e-07	385	5.95064e-08	618	3.77044e-08	851	100
153	3.38051e-07	386	100	619	4.07183e-08	852	100
154	2.12024e-07	387	78.8262	620	1.36882e-07	853	100
155	4.22157e-08	388	100	621	2.92982e-08	854	100
156	1.5468e-07	389	100	622	4.50559e-08	855	100
157	3.87516e-07	390	100	623	2.82885e-08	856	100
158	3.87516e-07	391	100	624	2.07988e-08	857	1.04596e-08
159	1.3831e-06	392	56.4784	625	1.77983e-08	858	100
160	4.79717e-07	393	100	626	1.9582e-08	859	100
161	49.0921	394	100	627	5.68782e-08	860	100
162	6.13672e-08	395	8.20026e-06	628	8.64019e-09	861	100
163	100	396	6.71269e-08	629	7.87476e-09	862	100
164	1.5468e-07	397	3.38579e-08	630	5.66023e-07	863	100
165	1.07222e-07	398	5.06326e-07	631	1.62355e-07	864	100
166	2.82807e-08	399	7.6426	632	1.09438e-08	865	100
167	5.14082e-08	400	3.06399e-07	633	1.14288e-08	866	100
168	6.5556e-08	401	6.73477e-08	634	6.08075e-08	867	100
169	1.99022e-07	402	10.9476	635	4.48262e-08	868	95.0609
170	1.55763e-07	403	10.2913	636	1.03931e-08	869	80.4406
171	9.57792e-08	404	55.463	637	4.10875e-07	870	1.7725e-08
172	1.77256e-07	405	55.463	638	1.45356e-08	871	100
173	1.28079e-07	406	100	639	9.97896e-09	872	100
174	1.28079e-07	407	100	640	7.40196e-08	873	100
175	1.5468e-07	408	5.92706e-08	641	4.53697e-08	874	100
176	3.76924e-07	409	16.7006	642	1.34486e-08	875	2.971e-08
177	7.17095e-08	410	95.4299	643	2.33863e-08	876	100
178	4.93011e-08	411	88.144	644	1.3594e-08	877	16.1071
179	4.09529e-08	412	2.30298e-07	645	1.98503e-08	878	1.77163e-08
180	4.39501e-08	413	2.48605e-08	646	3.63608e-08	879	77.6427
181	1.5468e-07	414	100	647	4.307e-08	880	100
182	1.84953e-07	415	44.7108	648	1.48356e-08	881	100

183	1.54054e-07	416	100	649	1.35881e-08	882	100
184	1.84953e-07	417	100	650	3.11491e-08	883	1.62455e-07
185	1.28079e-07	418	97.1921	651	1.15842e-08	884	29.8817
186	4.62787e-07	419	97.1921	652	6.74876	885	100
187	1.9273e-08	420	32.7374	653	9.52087e-08	886	71.0149
188	1.3828e-08	421	71.4222	654	7.36822	887	100
189	2.82894e-08	422	3.45104e-08	655	1.99873e-07	888	100
190	1.4286e-08	423	3.89171e-07	656	7.56789e-08	889	100
191	1.50038e-08	424	1.17789e-08	657	8.36207e-08	890	100
192	2.22259e-08	425	47.6717	658	4.73264e-08	891	100
193	2.30676e-08	426	100	659	9.26251	892	100
194	1.53127e-08	427	8.29243e-08	660	12.6838	893	56.862
195	1.63587e-08	428	3.64277e-08	661	1.7387e-07	894	100
196	6.7986e-08	429	3.52821e-08	662	86.5115	895	100
197	2.43525e-08	430	8.84289	663	1.08575e-07	896	100
198	5.32598e-08	431	7.09862e-08	664	91.7738	897	100
199	3.43494	432	6.81808e-07	665	35.1577	898	100
200	1.46164e-07	433	5.45843e-08	666	14.0485	899	100
201	7.88909e-08	434	2.80329e-08	667	84.9883	900	100
202	2.62519e-08	435	1.17527e-08	668	5.00021e-07	901	100
203	3.36905e-08	436	4.9578e-08	669	9.70082e-08	902	100
204	5.10527e-08	437	2.68206e-08	670	7.58446e-08	903	100
205	2.45367e-08	438	2.24286e-05	671	58.0712	904	100
206	3.07102e-08	439	7.62233e-07	672	6.29752e-08	905	100
207	11.3626	440	6.63937e-08	673	35.4276	906	100
208	4.64869e-08	441	2.48883e-08	674	4.33488e-06	907	3.30393
209	2.03694e-08	442	100	675	5.56842e-08	908	100
210	1.92825e-08	443	3.54369e-08	676	2.95101e-08	909	100
211	1.02081e-07	444	100	677	2.95615e-08	910	100
212	1.40133e-08	445	100	678	1.70771e-08	911	60.9945
213	34.3667	446	100	679	32.1771	912	95.8192
214	6.60261e-08	447	100	680	7.1615e-08	913	43.4258
215	4.53354e-08	448	62.7717	681	3.36267e-07	914	1.50456e-07
216	5.08046e-08	449	3.3631e-08	682	5.90314e-08	915	100
217	6.234e-08	450	6.61826e-09	683	7.40521e-08	916	19.1198
218	8.34583e-09	451	1.96369e-08	684	9.2942e-08	917	100
219	1.91146e-08	452	100	685	10.7575	918	55.288
220	2.72102e-08	453	6.45754e-08	686	59.5014	919	100
221	1.2318e-08	454	7.66703e-07	687	19.0044	920	100
222	2.73081e-08	455	4.03601e-08	688	9.32704e-07	921	100
223	2.45947e-08	456	1.64845e-07	689	1.60998e-07	922	100
224	5.64483e-08	457	71.0304	690	82.9783	923	100
225	2.69912e-08	458	4.19629e-08	691	2.57498e-08	924	100
226	1.63384e-08	459	3.58107e-08	692	3.11657e-08	925	100
227	7.12442e-09	460	70.1068	693	9.5684	926	100
228	1.65566e-08	461	100	694	9.31443e-08	927	100

```
229   8.65693e-09   462   2.9108e-08    695   1.26076e-07   928   4.83518e-08
230   1.14368e-08   463   1.4836e-07    696   1.39693e-08   929   3.88496e-07
231   2.09631e-08   464   3.21062e-08   697  40.7794        930  36.1626
232   1.90661e-08   465   1.93438e-07   698  10.2959
233  18.3483        466   3.24508e-08   699   3.94902e-08
;
```

The above results contain 155 support vectors from among the 930 input data elements. This relatively low percentage of the total input data elements suggests that the choice of $C = 100$ was a reasonable one. Calculating the b value for each support vectors verifies that we get the same value for each.

```
#alpha index, alpha value, calculated b
3 5.7145 1.070456954818
5 7.0532 1.070458003550
15 29.3529 1.070456917860
19 5.7228 1.070457321617
24 87.9023 1.070456625551
25 54.8041 1.070457086784
52 31.0034 1.070457811208
60 10.4148 1.070456139039
92 1.3167 1.070456399823
105 32.7716 1.070456500112
116 11.9145 1.070456581841
124 47.6849 1.070456477399
136 54.0187 1.070455837201
140 19.8018 1.070456446670
144 4.4246 1.070456433282
145 4.4246 1.070456433282
146 2.2519 1.070456667867
160 49.0921 1.070456525914
198 3.4349 1.070456384679
206 11.3626 1.070458249808
212 34.3667 1.070456498634
232 18.3483 1.070456385869
236 1.5172 1.070456647091
276 28.6054 1.070456057211
279 42.4313 1.070456639370
282 31.6708 1.070456735161
288 10.9079 1.070455501098
305 0.1751 1.070381117977
307 9.9847 1.070456389796
312 28.4913 1.070455653409
314 20.7138 1.070456614663
```

316	55.0932	1.070457995762
323	3.6713	1.070456015144
329	0.6377	1.070456248119
337	26.5263	1.070457614809
340	76.5891	1.070457006468
347	50.2192	1.070456808547
359	3.0698	1.070456029499
360	18.6384	1.070455919222
361	6.1922	1.070456396369
362	78.9539	1.070455745002
372	57.4036	1.070456414550
373	80.5618	1.070456775276
376	7.8096	1.070456150539
379	21.3358	1.070457653196
380	3.3469	1.070456700692
382	19.7623	1.070456114907
386	78.8262	1.070456183524
391	56.4784	1.070457004452
398	7.6426	1.070456665992
401	10.9476	1.070457026473
402	10.2913	1.070456211710
403	55.4630	1.070456494051
404	55.4630	1.070456494051
408	16.7006	1.070456330389
409	95.4299	1.070456683881
410	88.1440	1.070457111230
414	44.7108	1.070457330137
417	97.1921	1.070456082030
418	97.1921	1.070456082030
419	32.7374	1.070455447269
420	71.4222	1.070456351084
424	47.6717	1.070456308539
429	8.8429	1.070456125046
447	62.7717	1.070457240982
456	71.0304	1.070456527582
459	70.1068	1.070456765760
467	19.8573	1.070456469188
473	0.3536	1.070453295132
477	17.6082	1.070455989484
484	3.5712	1.070456784877
488	36.5089	1.070456801288
496	0.5172	1.070455405047
508	15.1377	1.070455287170
520	12.4197	1.070457034677
527	37.3450	1.070455997986
528	5.2123	1.070456714431

529	29.8444	1.070456600401
533	70.0837	1.070456363542
538	72.0049	1.070456247672
545	5.1858	1.070456556141
552	14.6866	1.070456764543
553	51.7308	1.070456424819
554	20.6688	1.070456232390
555	4.8864	1.070456320348
563	19.1401	1.070457944652
574	75.4697	1.070456477919
651	6.7488	1.070456447351
653	7.3682	1.070456680096
658	9.2625	1.070456497349
659	12.6838	1.070456515841
661	86.5115	1.070456481616
663	91.7738	1.070456456564
664	35.1577	1.070456411878
665	14.0485	1.070456610638
666	84.9883	1.070456200115
670	58.0712	1.070457679985
672	35.4276	1.070456687916
678	32.1771	1.070456718883
684	10.7575	1.070456253366
685	59.5014	1.070456122071
686	19.0044	1.070456369104
689	82.9783	1.070456075705
692	9.5684	1.070456764190
696	40.7794	1.070456801773
697	10.2959	1.070457073814
699	18.4227	1.070456614667
711	75.9297	1.070456463361
715	18.5722	1.070456524221
720	7.0089	1.070457290112
725	13.7361	1.070456591122
727	8.6543	1.070456620797
729	51.0242	1.070456358756
734	40.0125	1.070455959872
738	12.0501	1.070456449213
743	24.7776	1.070457480324
752	63.8287	1.070455529656
762	30.3426	1.070457621112
767	98.6327	1.070456496127
769	27.4658	1.070457317837
774	39.8487	1.070454592313
775	80.8929	1.070457578619
784	28.4671	1.070455676275

Table 5: Dual Polynomial All Digits Error

Data Set	Error	95% Confidence Interval	
		Lower Bound	Upper Bound
Training	0.029	0.018	0.040
Testing	0.200	-0.161	0.239

```
789 13.6659 1.070455131373
793 1.5172 1.070456647091
797 11.4375 1.070457837635
801 43.5287 1.070457748747
804 33.8906 1.070456243569
810 6.9482 1.070456603963
812 32.9226 1.070456314772
813 24.7422 1.070455201870
814 38.3727 1.070458871934
815 88.5056 1.070456433800
816 18.4125 1.070455626052
826 64.2707 1.070456508596
827 10.2150 1.070455437992
835 45.5049 1.070454735535
839 75.5841 1.070456811927
841 1.7580 1.070457939916
842 68.7307 1.070457902706
845 96.2966 1.070456418024
867 95.0609 1.070455977351
868 80.4406 1.070456095607
876 16.1071 1.070455674830
878 77.6427 1.070457079420
883 29.8817 1.070455733096
885 71.0149 1.070456266973
892 56.8620 1.070457034931
906 3.3039 1.070457104102
910 60.9945 1.070456691140
911 95.8192 1.070455688802
912 43.4258 1.070458233218
915 19.1198 1.070456181082
917 55.2880 1.070457362156
929 36.1626 1.070455834797
```

As indicated in Table 5, the overall testing misclassification error achieved by the polynomial SVM classifier was 0.2. This is significantly better than the

0.9 misclassification error that we would expect to achieve by random guessing.