CS 440/ECE 448 Artificial Intelligence
Assignment 3: Naive Bayes Classification

Haoen CUI[*], *Guohao (Holden) DOU*[†], *Chuchao LUO*[‡]

November 27, 2017

Contents

[*]Haoen CUI’s Email hcui10@illinois.edu
[†]Guohao (Holden) DOU’s Email gdou2@illinois.edu
[‡]Chuchao LUO’s Email chuchao2@illinois.edu

1 Part 1: Digit Classification

1.1 Single Pixels as Features (For Everybody)

1.1.1 Implementation

We treated this problem as a special case of *pixel group as features* where the pixel groups are simply disjoint and of size 1 by 1. Please see the next section for details (e.g choice of Laplace smoothing constant, classification rates, and confusion matrices).

1.1.2 Posterior Probabilities: Highest and Lowest

Test examples with largest and smallest posterior probabilities for each digit are shown below.

```
## [1] " '======================== Digit = 0 ========================',"
## [1] " '---- Highest Posterior ------------- Lowest Posterior -----',"
## [1] " ['                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '              ####           |                             ',"
## [1] "  '            #######          |         ##########          ',"
## [1] "  '           ##########        |        ############         ',"
## [1] "  '          ############       |      ###############        ',"
## [1] "  '         ######## ####       |     #################       ',"
## [1] "  '        ########  #####      |     ##########  ######      ',"
## [1] "  '        #####      ####      |     ########      ####      ',"
## [1] "  '       #####        ###      |       #####       #####     ',"
## [1] "  '       #####        ###      |       ####         #####    ',"
## [1] "  '       ####         ###      |       ####          ####    ',"
## [1] "  '       ####         ###      |       ####          ####    ',"
## [1] "  '       ####         ###      |       ####          ####    ',"
## [1] "  '       ####        ####      |       ####          ####    ',"
## [1] "  '       ####       ####       |       #####         ####    ',"
## [1] "  '       #####     #####       |       ######        ####    ',"
## [1] "  '       ######   ######       |        #####        ####    ',"
## [1] "  '        #############        |         ######      ####    ',"
## [1] "  '        ############         |          ###############    ',"
## [1] "  '         ##########          |           ##############    ',"
## [1] "  '           ######            |            ############     ',"
## [1] "  '                             |             ##########      ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ']"
## [1] " '======================== Digit = 1 ========================',"
## [1] " '---- Highest Posterior ------------- Lowest Posterior -----',"
## [1] " ['                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |            ###              ',"
## [1] "  '                             |            ####             ',"
## [1] "  '              ###            |            ####             ',"
## [1] "  '              ####           |            ####             ',"
## [1] "  '              ####           |            ####             ',"
```

```
## [1] "  '              ####           |            ####             ',"
## [1] "  '              ####           |            ####             ',"
## [1] "  '              ###            |            ####             ',"
## [1] "  '             ####            |            ####             ',"
## [1] "  '             ####            |           ######            ',"
## [1] "  '             ####            |           ######            ',"
## [1] "  '             ####            |           ## ###            ',"
## [1] "  '             ###             |              ####           ',"
## [1] "  '            ####             |               ####          ',"
## [1] "  '            ####             |               ####          ',"
## [1] "  '            ####             |                #####        ',"
## [1] "  '           ####              |          ###########        ',"
## [1] "  '           ####              |         ############        ',"
## [1] "  '          #####              |         ############        ',"
## [1] "  '          #####              |          ##########         ',"
## [1] "  '          #####              |                             ',"
## [1] "  '           ###               |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ']"
## [1] "'======================== Digit = 2 ========================',"
## [1] "'---- Highest Posterior ------------- Lowest Posterior -----',"
## [1] " ['                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '          ######             |                             ',"
## [1] "  '          ########           |                             ',"
## [1] "  '         ##########          |                             ',"
## [1] "  '         ####  #####         |                             ',"
## [1] "  '         ##     ####         |              #########      ',"
## [1] "  '                 ###         |         ##############      ',"
## [1] "  '                 ###         |       #################     ',"
## [1] "  '                  ##         |     ###################     ',"
## [1] "  '                  ##         |    ####################     ',"
## [1] "  '                  ##         |    ###########  #######     ',"
## [1] "  '                 ###         |    ##########  #######      ',"
## [1] "  '                 ###         |    ####      ########       ',"
## [1] "  '           ##### ###         |    ###      #########       ',"
## [1] "  '         ##########          |    ###     ##########       ',"
## [1] "  '        ###########          |    ##      ########         ',"
## [1] "  '        #############        |            ########         ',"
## [1] "  '        ###  ###########     |            #########        ',"
## [1] "  '        #########  ####      |            ###########      ',"
## [1] "  '        ########             |             ###########     ',"
## [1] "  '         #####               |               #########     ',"
## [1] "  '                             |                #######      ',"
## [1] "  '                             |                   ####      ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ']"
## [1] "'======================== Digit = 3 ========================',"
## [1] "'---- Highest Posterior ------------- Lowest Posterior -----',"
## [1] " ['                             |                             ',"
```

```
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '         ########            |                             ',"
## [1] "  '         ##########          |                             ',"
## [1] "  '            ########         |                             ',"
## [1] "  '                ####         |          ############       ',"
## [1] "  '                 ####        |        ###############      ',"
## [1] "  '                ####         |        ################     ',"
## [1] "  '              #####          |      #####         ####     ',"
## [1] "  '             #####           |      ####          ####     ',"
## [1] "  '            #####            |      ###          ####      ',"
## [1] "  '           #######           |                  ####       ',"
## [1] "  '           ########          |                 #####       ',"
## [1] "  '           ##  ####          |               ######        ',"
## [1] "  '                ###          |               #####         ',"
## [1] "  '                ###          |               #####         ',"
## [1] "  '                ###          |                 ###         ',"
## [1] "  '               ####          |                ###          ',"
## [1] "  '              #####          |              #####          ',"
## [1] "  '     #############           |             ######          ',"
## [1] "  '     ############            |            ######           ',"
## [1] "  '      ##########             |           #####             ',"
## [1] "  '                             |         #######             ',"
## [1] "  '                             |        #######              ',"
## [1] "  '                             |        #####                ',"
## [1] "  '                             |                             ']"
## [1] " '======================== Digit = 4 ========================='," 
## [1] " '---- Highest Posterior ------------- Lowest Posterior -----',"
## [1] " ['                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |        ###                  ',"
## [1] "  '           #       ##        |        ###                  ',"
## [1] "  '          ###      ##        |        ###    ####          ',"
## [1] "  '          ###      ##        |        ###    ####          ',"
## [1] "  '         ####    ####        |       ####    ####      #   ',"
## [1] "  '        ####     ####        |       ####    ####     ###  ',"
## [1] "  '        ####     ####        |       ####    ####     ###  ',"
## [1] "  '        ###      ####        |       ####    ####     ###  ',"
## [1] "  '        ###     #####        |       ####    ####    ####  ',"
## [1] "  '        ##    #######        |       ####     ###  ######  ',"
## [1] "  '      #############          |       ####     ##########   ',"
## [1] "  '       ############          |       #### #############    ',"
## [1] "  '          ########           |       ################      ',"
## [1] "  '              ###            |       ##############        ',"
## [1] "  '              ###            |         ###########         ',"
## [1] "  '              ###            |                 ####        ',"
## [1] "  '             ####            |                 ####        ',"
## [1] "  '             ###             |                 ####        ',"
## [1] "  '             ###             |                  ###        ',"
## [1] "  '             ###             |                   ##        ',"
```

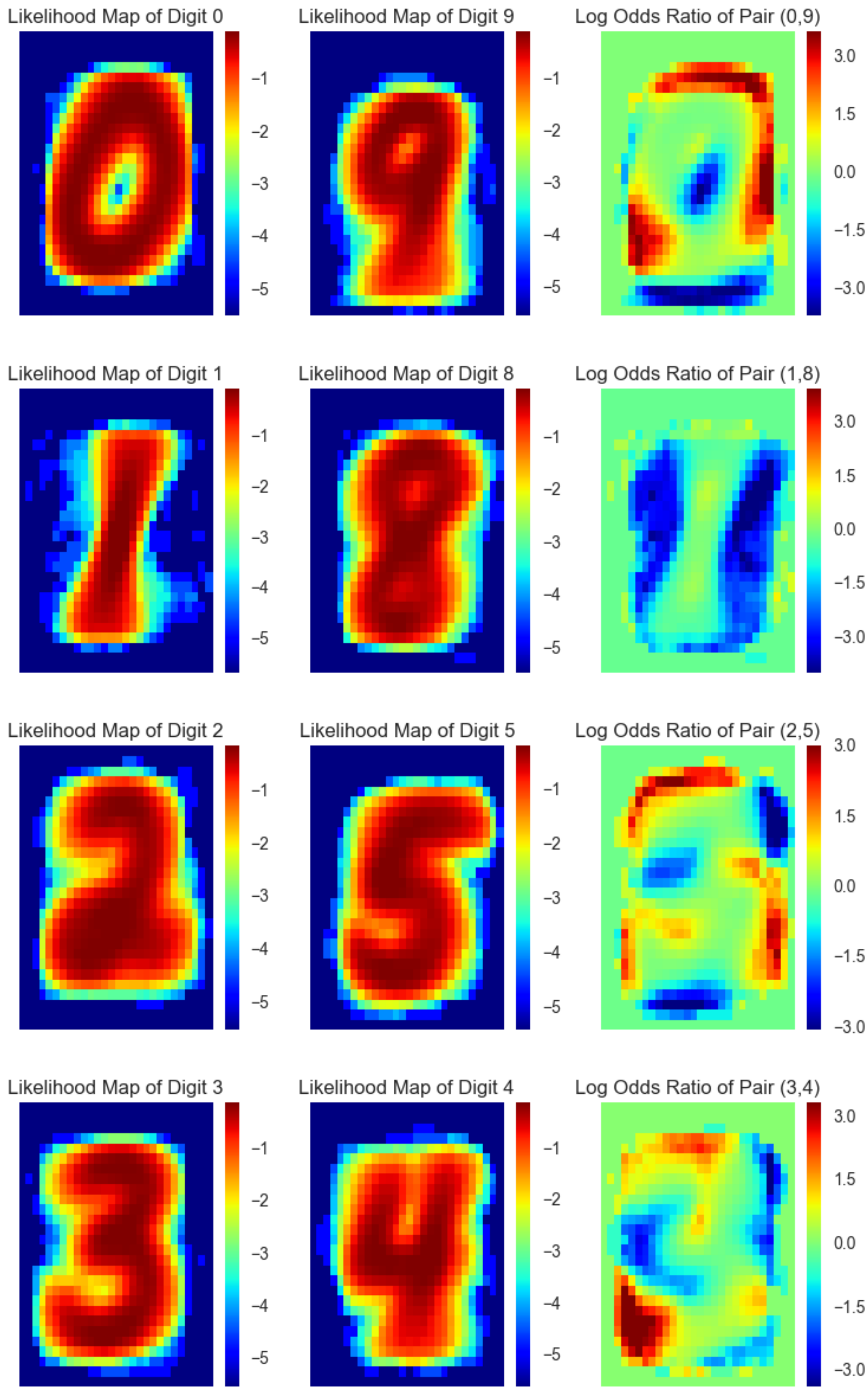
```
## [1] "  '             ###             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ']"
## [1] "'======================== Digit = 5 ========================',"
## [1] "'---- Highest Posterior ------------- Lowest Posterior -----',"
## [1] " ['                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                   ####      |       #####                 ',"
## [1] "  '            ###########      |      #######                ',"
## [1] "  '            ##########       |      #######                ',"
## [1] "  '            ########         |      #######                ',"
## [1] "  '           ####              |      ######                 ',"
## [1] "  '           ###               |      #######                ',"
## [1] "  '          ###                |       #########             ',"
## [1] "  '          ######             |           #######           ',"
## [1] "  '         ########            |              ######         ',"
## [1] "  '         ########            |                #####        ',"
## [1] "  '          #    ###           |                   ####      ',"
## [1] "  '               ###           |                    ####     ',"
## [1] "  '               ###           |                     ####    ',"
## [1] "  '                ##           |            ####     #####   ',"
## [1] "  '       ###      ###          |            ##############   ',"
## [1] "  '       ###     ###           |             #############   ',"
## [1] "  '       ###     ###           |              ############   ',"
## [1] "  '       ####   ####           |                   ######    ',"
## [1] "  '        #########            |                             ',"
## [1] "  '           #####             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ']"
## [1] " '======================== Digit = 6 ========================',"
## [1] " '---- Highest Posterior ------------- Lowest Posterior -----',"
## [1] " ['                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                ###          |        ###                  ',"
## [1] "  '              #####          |        ####                 ',"
## [1] "  '             #####           |       ####                  ',"
## [1] "  '            #####            |       ####                  ',"
## [1] "  '            #####            |       ###                   ',"
## [1] "  '           #####             |       ###        ###        ',"
## [1] "  '          #####              |       ###        #####      ',"
## [1] "  '          ####               |      ####       ######      ',"
## [1] "  '          ####               |      ####       ######      ',"
## [1] "  '         #####               |      ####       #######     ',"
## [1] "  '         ####                |      ###        #######     ',"
## [1] "  '         ####   #####        |      ###        #######     ',"
## [1] "  '        #####  ######        |      ####       #######     ',"
## [1] "  '        #############        |      ####       #######     ',"
## [1] "  '        #### ########        |       ####      ######      ',"
## [1] "  '        #############        |       ####       #####      ',"
```

```
## [1] "  '        ###########          |        ####      ####       ',"
## [1] "  '         #########           |         ##############      ',"
## [1] "  '          #######            |           ##########        ',"
## [1] "  '            ####             |            #######          ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ']"
## [1] "'======================== Digit = 7 ========================',"
## [1] "'---- Highest Posterior ------------- Lowest Posterior -----',"
## [1] " ['                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |           ######    ####    ',"
## [1] "  '        #### ######          |           ###############   ',"
## [1] "  '        ############         |           ###############   ',"
## [1] "  '        ############         |            ##############   ',"
## [1] "  '          ###    ###         |                  ######     ',"
## [1] "  '                ####         |                 #######     ',"
## [1] "  '                ####         |                 #####       ',"
## [1] "  '               ####          |               ######        ',"
## [1] "  '               ####          |          ##########         ',"
## [1] "  '               ####          |         ##########          ',"
## [1] "  '              ####           |         #############       ',"
## [1] "  '              ####           |         #############       ',"
## [1] "  '              ###            |         #############       ',"
## [1] "  '             ####            |        ######               ',"
## [1] "  '             ###             |        #####                ',"
## [1] "  '            ####             |       #####                 ',"
## [1] "  '            ###              |      #####                  ',"
## [1] "  '           ###               |      #####                  ',"
## [1] "  '           ###               |      #####                  ',"
## [1] "  '           ###               |      ###                    ',"
## [1] "  '           ###               |                             ',"
## [1] "  '                             |                             ']"
## [1] "'======================== Digit = 8 ========================',"
## [1] "'---- Highest Posterior ------------- Lowest Posterior -----',"
## [1] " ['                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                ####         |          #####              ',"
## [1] "  '              #######        |        #########            ',"
## [1] "  '            #########        |      ###########            ',"
## [1] "  '           ##########        |      #############          ',"
## [1] "  '          ###### ####        |      ############# #        ',"
## [1] "  '          ####   ####        |      ################       ',"
## [1] "  '          ###     ####       |      ######   #######       ',"
## [1] "  '          ###    #####       |       ##############        ',"
## [1] "  '          ############       |       ############          ',"
```

```
## [1] "  '          ##########         |        ###########          ',"
## [1] "  '           #######           |         ###########         ',"
## [1] "  '          ######             |          #############      ',"
## [1] "  '         ########            |          #############      ',"
## [1] "  '         ########            |          ###############    ',"
## [1] "  '        ####  ####           |            ##############   ',"
## [1] "  '        ####  ####           |            ##############   ',"
## [1] "  '        #### ####            |             #############   ',"
## [1] "  '        #########            |               ###########   ',"
## [1] "  '         ########            |                ##########   ',"
## [1] "  '          ######             |                   #####     ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ']"
## [1] "'======================== Digit = 9 ========================',"
## [1] "'---- Highest Posterior ------------- Lowest Posterior -----',"
## [1] " ['                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |                             ',"
## [1] "  '                             |               #######       ',"
## [1] "  '            ######           |           ############      ',"
## [1] "  '           ########          |         ###############     ',"
## [1] "  '         ###########         |      ##################     ',"
## [1] "  '        #####   ####         |     ############## ####     ',"
## [1] "  '        ####   #####         |    ######           ###     ',"
## [1] "  '       ####    #####         |    #####            ###     ',"
## [1] "  '       ####    #####         |    ####           #####     ',"
## [1] "  '       ####  ######          |    ####        ########     ',"
## [1] "  '        ###########          |    ####################     ',"
## [1] "  '         #########           |    ####################     ',"
## [1] "  '          ########           |     #############  ####     ',"
## [1] "  '             ####            |       ########     ####     ',"
## [1] "  '             ####            |                    ####     ',"
## [1] "  '             ###             |                    ####     ',"
## [1] "  '            ####             |                    ####     ',"
## [1] "  '            ####             |                    ####     ',"
## [1] "  '            ###              |                    ####     ',"
## [1] "  '            ###              |                    ####     ',"
## [1] "  '            ###              |                    ####     ',"
## [1] "  '            ###              |                             ',"
## [1] "  '                             |                             ']"
```

1.1.3 Visualization of Likelihoods and Odds Ratios

We interpreted `pairs of digits that have the highest confusion rates` as `pairs without misclassifications`. Hence, we chose pairs `(0, 9)`, `(1, 8)`, `(2, 5)`, and `(3, 4)` (randomly out of all eligible pairs). The plots are shown below. We tried several different color maps, but none of them match exactly with the example on the assignment page. The following are the closest version we can get. Though the color maps are different, they convey the same information. Also, note that the smoother used to generate these results will be described as a special case with disjoint kernel size `(1, 1)` in the next section.



1.2 Pixel Groups as Features (For Four-Credit Students)

1.2.1 Implementation

1.2.2 Choice of Smoothing Constant: 10-Fold Cross Validation

We used 10-fold cross validation to select the smoothing constant. To be specific,

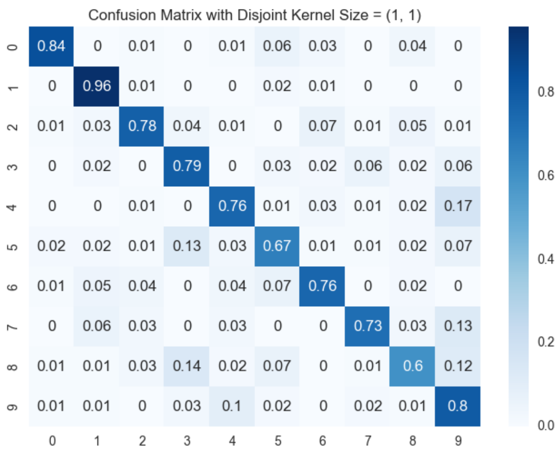
- We randomly assigned a fold number (out of 1 to 10) to each record in the training set.
- Then, we iterate through all potential smoothers on the each fold.
- For a given smoother and a selected fold number, the selected fold will serve as the validation set and the remaining training set will be used to train the Naive Bayes model. We can calculate a performance measure (in this case: overall misclassification rate). Thus, we will get 10 measures for each smoother.
- We summarized the performance of each smoother using the average of its 10 measures.
- Finally, we selected the best smoother based on the aggregated averages.
- **IMPORTANT NOTICE**: testing set is not being used in the entire cross validation phase. It is only used once for testing purposes. Relevant results (e.g. plots and prototypicals) are generated based on these tests.

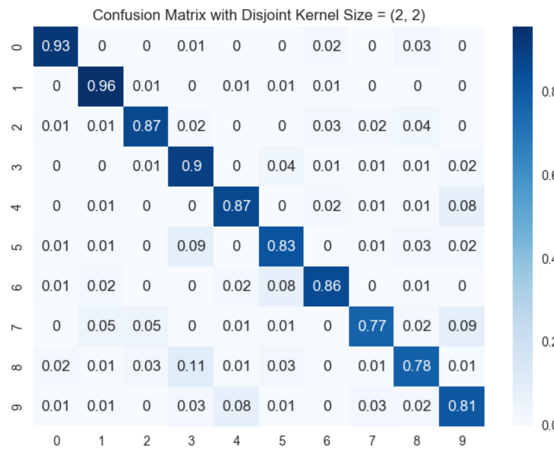
We considered smoothers in the list `smoothers = [0.1, 0.5, 1, 2, 4, 8]`. The best smoothers selected for different kernel sizes are shown below. The entire assignment used the same cross-validation methodology. Hence, we will not repeat this section again.

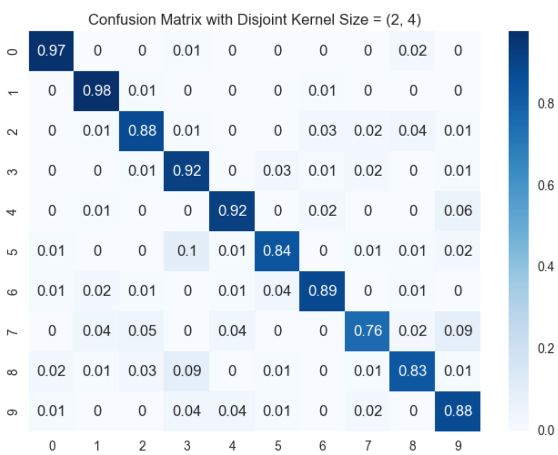
Table 1: Choice of Smoothing Constant Using 10-Fold Cross Validation

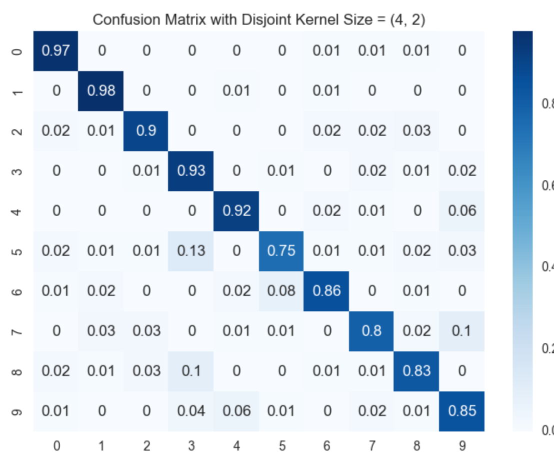
Kernel Type	Kernel Size	Best Smoother Value
Disjoint	(1, 1)	0.5
Disjoint	(2, 2)	0.1
Disjoint	(2, 4)	0.1
Disjoint	(4, 2)	0.1
Disjoint	(4, 4)	0.1
Overlapping	(2, 2)	0.1
Overlapping	(2, 4)	0.1
Overlapping	(4, 2)	0.1
Overlapping	(4, 4)	0.1
Overlapping	(2, 3)	0.1
Overlapping	(3, 2)	0.1
Overlapping	(3, 3)	0.1

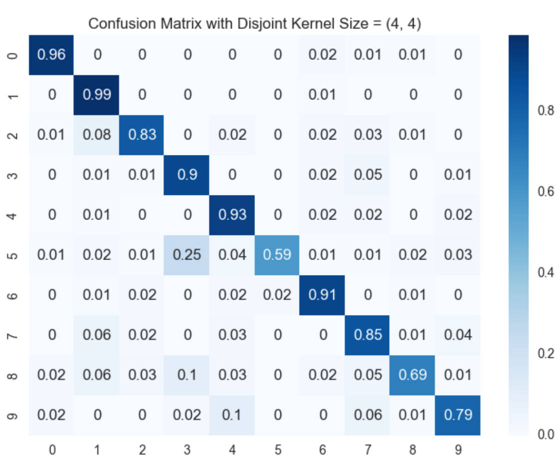
1.2.3 Accuracy on Test Set: Classification Rate and Confusion Matrix

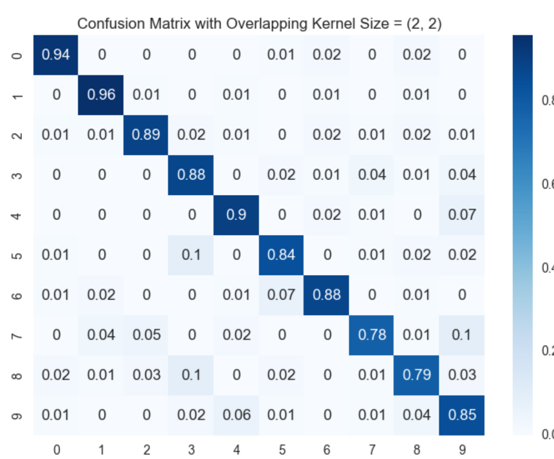












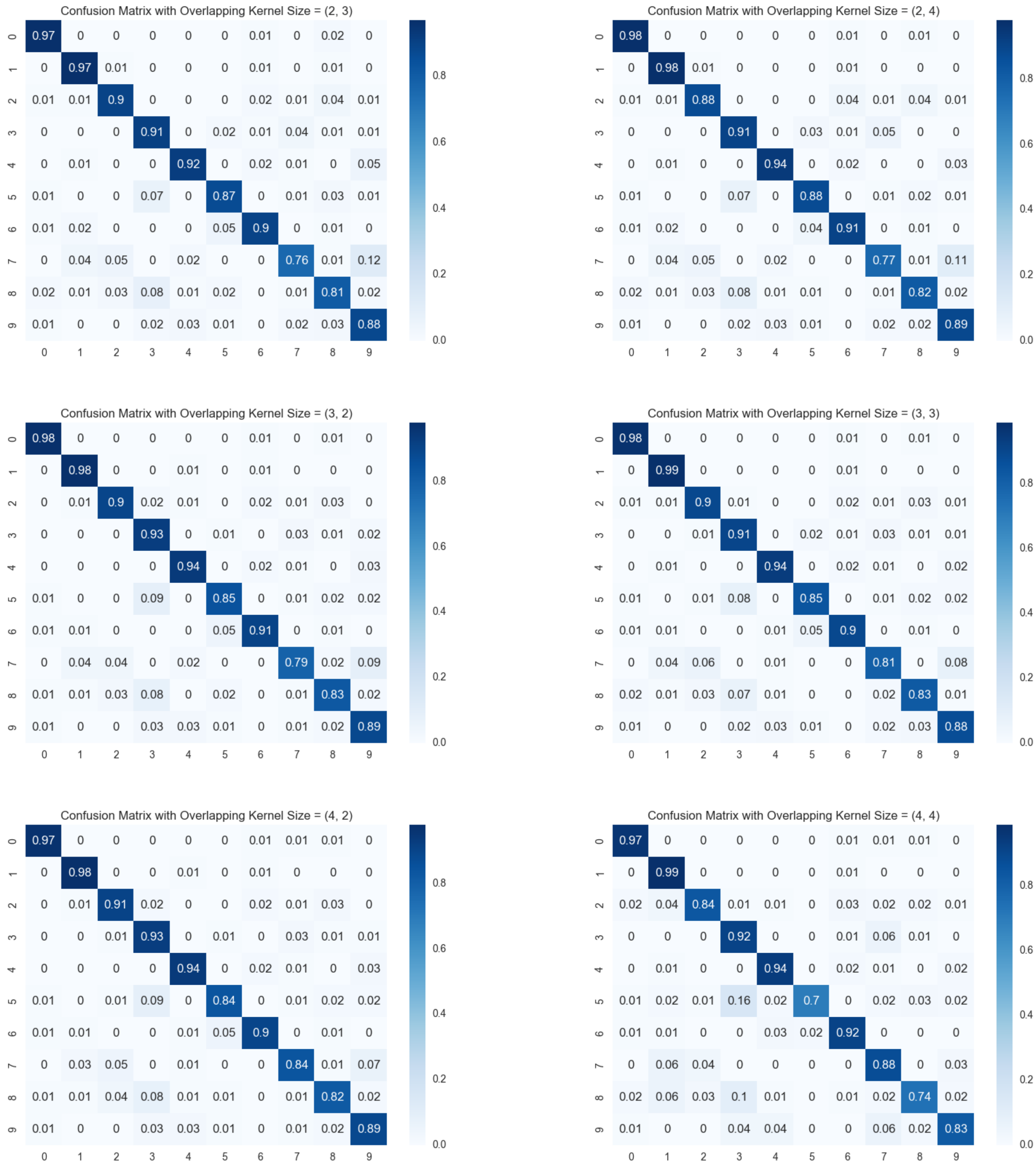


Table 2: Overall Accuracy on Different Kernels with Best Smoothers

Kernel Type	Kernel Size	Overall Accuracy
Disjoint	(1, 1)	77.0 %
Disjoint	(2, 2)	85.8 %
Disjoint	(2, 4)	88.6 %

Kernel Type	Kernel Size	Overall Accuracy
Disjoint	(4, 2)	87.9 %
Disjoint	(4, 4)	84.6 %
Overlapping	(2, 2)	87.1 %
Overlapping	(2, 4)	89.6 %
Overlapping	(4, 2)	90.2 %
Overlapping	(4, 4)	87.4 %
Overlapping	(2, 3)	88.8 %
Overlapping	(3, 2)	90.0 %
Overlapping	(3, 3)	90.0 %

1.2.4 Trends for Different Feature Sets

The following are some general trends we found,

- with same kernel size, overlapping kernels tend to perform better than disjoint kernels because overlapping ones contain more features.

- with the same kernel type, increase in kernel size does not necessary translate to better performance because at some point using additional features may be merely overfitting.

- by trying out different kernels and tuning the smoother, one can achieve a much higher performance than merely using the default 1 by 1 pixels.

1.2.5 Running Time for Different Feature Sets

1.2.5.1 Training

1.2.5.2 Testing

1.3 Extra Credit

1.3.1 Ternary Features

This part is relatively easy. We merely changed the way we read in files to incorporate ternary features. Then, we run the same process as the one we used for binary features with disjoint 1 by 1 kernel. Finally, we ended up with the same best smoother (`= 0.5`) and the resulting overall classification rate is 77.1 %. There is not much improvements.

1.3.2 Naive Bayes Classifier on Face Data

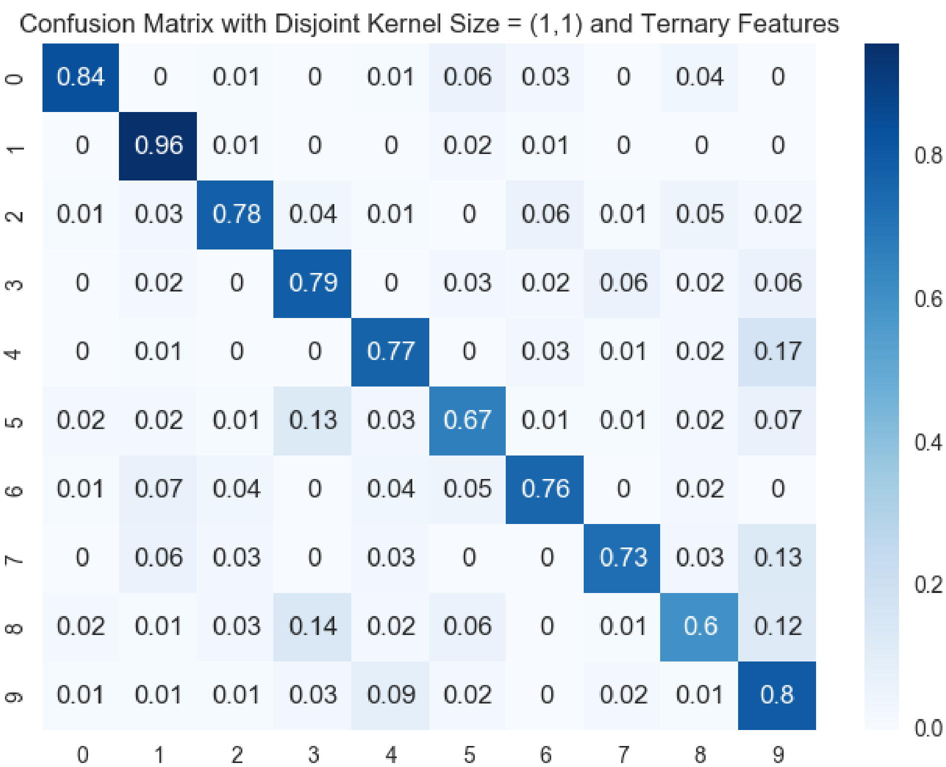


Figure 1: Ternary Confusion Matrix

2 Part 2: Audio Classification

2.1 Binary Classification: Hebrew Words of “Yes” and “No” (For Everybody)

2.1.1 Implementation

2.1.2 Classification Rate and Confusion Matrix

2.2 Multi-Class Classification: Audio Digits 1-5 Spoken by Four Different Speakers (For Four-Credit Students)

2.2.1 Implementation

2.2.2 Overall Accuracy

2.2.3 Classification Rate and Confusion Matrix

2.3 Extra Credit

2.3.1 Binary Classification on Unsegmented Data

2.3.2 Alternative Method (RNN) on XXX Data

2.3.3 Average-Column Method on Hebrew Yes-No Corpus

---

3 Statement of Individual Contribution

Table 3: Statement of Individual Contribution

	NetID	Contribution
Haoen CUI	hcui10	visualization, report, and ideas generation
Guohao DOU	gdou2	part 1 (algorithm design and programming) and ideas generation
Chuchao LUO	chuchao2	part 2 (algorithm design and programming) and ideas generation