CSCI567 2013 Homework Assignment 2

Programming Report

Student Name	Last 4 digits of USC ID
Jia Li	0854
Udit Agrawal	5165
Yin-Ray Rick Huang	6794
Arpit Bansal	0979

1 Digit Classification using K-Nearest Neighbors

1.1 Question 7

$p = 0.5$

	1	**3**	**5**	**7**	**9**	**11**	**13**	**15**	**17**	**19**
Training Acc (%)	66.46	68.02	70.58	71.42	71.98	72.56	72.82	73.06	73.00	73.12
Development Acc(%)	78.72	78.92	80.47	80.27	80.92	80.92	80.75	80.55	80.55	80.62
Test Acc(%)	78.80	78.20	78.75	79.35	78.90	79.10	79.35	79.00	79.15	78.75

$p = 1$

	1	**3**	**5**	**7**	**9**	**11**	**13**	**15**	**17**	**19**
Training Acc (%)	70.24	72.42	74.68	75.64	75.86	76.38	76.80	76.50	76.66	76.52
Development Acc(%)	82.23	82.83	83.83	83.97	83.67	83.75	83.47	82.93	83.10	82.88
Test Acc(%)	83.15	81.65	82.55	82.75	82.50	82.70	82.20	81.40	81.75	80.95

$p = 1.5$

	1	**3**	**5**	**7**	**9**	**11**	**13**	**15**	**17**	**19**
Training Acc (%)	71.64	73.90	75.78	76.82	77.68	77.66	77.86	77.66	77.50	77.96
Development Acc(%)	83.53	84.00	84.55	84.62	84.60	84.20	84.28	84.12	84.05	83.85
Test Acc(%)	83.55	83.35	84.15	84.15	83.60	83.80	83.25	82.90	83.10	82.35

$p = 2$

	1	**3**	**5**	**7**	**9**	**11**	**13**	**15**	**17**	**19**
Training Acc (%)	72.90	74.50	76.96	77.94	78.20	78.66	78.64	78.46	78.28	78.26
Development Acc(%)	84.38	84.30	84.80	84.88	84.80	84.70	85.15	84.38	84.20	84.33
Test Acc(%)	83.40	84.00	84.60	84.00	84.00	83.80	83.60	83.10	82.70	83.05

$p = 3$

	1	**3**	**5**	**7**	**9**	**11**	**13**	**15**	**17**	**19**
Training Acc (%)	74.10	74.96	77.54	78.40	78.28	78.20	78.76	78.98	78.92	78.64
Development Acc(%)	83.97	84.28	84.55	84.95	85.22	84.95	84.82	84.20	84.30	83.95
Test Acc(%)	83.45	83.80	83.95	84.10	84.60	83.85	83.60	83.05	82.65	82.50

$p = \infty$

	1	**3**	**5**	**7**	**9**	**11**	**13**	**15**	**17**	**19**
Training Acc (%)	10.08	10.00	10.00	10.00	10.00	10.00	10.00	10.00	10.00	10.00
Development Acc(%)	13.88	10.50	10.08	10.00	10.00	10.00	10.00	10.00	10.00	10.00
Test Acc(%)	14.65	10.60	10.00	10.00	10.00	10.00	10.00	10.00	10.00	10.00

The K that maximizes the training set accuracy and the K that maximizes the development set accuracy.

	$p = 0.5$	$p = 1$	$p = 1.5$	$p = 2$	$p = 3$	$p = \infty$
Training Set	19	13	19	11	15	1
Developing Set	9 & 11	7	7	13	9	1

1.2 Question 8

$p = 3, k = 15$ maximize the training set accuracy.
$p = 3, k = 9$ maximize the developing set accuracy.

1.3 Question 9

For tuning the hyper parameter on training set, we need to use cross validation, and leave one out strategy can help us maximize the usage of training data set. Also it is good to use leave one out in KNN case, because the model is the training data itself, so there is no training process actually. Leave one out will take the same time as the K-fold cross validation.
Another thing to take care is, when you calculate the distance from a sample to all the others in the training set to find K nearest one, you should temporally remove it self from the training set. OR we will always get the 100 accuracy with k = 1, as its distance to itself is 0.

1.4 Question 10

First, we should never tuning the hyper parameter using testing set, as in real case the test data may not be available. Testing set is only used for reporting the result. Second, as long as we have the extra development set to tuning the hyper parameter, we do not need to do it on training set. Using cross validation is because we don't have enough data to use as development set. Also in each combination in { trainingSet testingSet } in cross validation, the trainingSet is not the full set of training data, so it cannot really represent the real case.
One most important thing here is that our training and teasing data seems from different source, as we can see the difference of accuracy for training, developing and testing (training from one source, developing and testing from another source). So it we using training set to find optimal K, it will more likely to fit the training data, not the testing data, kind of over fitting.

1.5 Question 11

1.5.1 Interesting Distance 1

Distance Function:

$d(\boldsymbol{U}, \boldsymbol{V}) = \sum_i (\boldsymbol{U}_i - \boldsymbol{V}_i \leq t)$ where t is a threshold. If for a corresponding element of two vectors, their difference is smaller than t, then count 1, or as zero. After that, sum all the dimensions, and normalized by D.

$t = 100$.

	1	**3**	**5**	**7**	**9**	**11**	**13**	**15**	**17**	**19**
Training Accuracy(%)	0.55	0.56	0.60	0.62	0.63	0.64	0.64	0.65	0.65	0.65
Developing Accuracy(%)	0.72	0.73	0.75	0.76	0.77	0.77	0.77	0.77	0.77	0.77
Testing Accuracy(%)	0.73	0.73	0.74	0.74	0.74	0.74	0.74	0.74	0.74	0.74

$t = 50$.

	1	**3**	**5**	**7**	**9**	**11**	**13**	**15**	**17**	**19**
Training Accuracy(%)	0.51	0.51	0.54	0.56	0.57	0.58	0.58	0.60	0.60	0.60
Developing Accuracy(%)	0.67	0.68	0.70	0.71	0.71	0.71	0.72	0.72	0.72	0.72
Testing Accuracy(%)	0.67	0.67	0.68	0.70	0.70	0.70	0.70	0.70	0.71	0.71

$t = 20$.

	1	**3**	**5**	**7**	**9**	**11**	**13**	**15**	**17**	**19**
Training Accuracy(%)	0.48	0.47	0.52	0.54	0.54	0.55	0.56	0.56	0.56	0.57
Developing Accuracy(%)	0.62	0.62	0.64	0.66	0.67	0.67	0.67	0.68	0.68	0.68
Testing Accuracy(%)	0.62	0.62	0.64	0.64	0.65	0.65	0.66	0.67	0.67	0.66

Code : the code is run_1_KNN_interesting_1(K, t, set). Where the t is the threshold.

1.5.2 Interesting Distance 2

Distance Function:

$d(\boldsymbol{U}, \boldsymbol{V}) = \frac{(\boldsymbol{U}-128*\mathbf{1})^T(\boldsymbol{V}-128*\mathbf{1})}{|\boldsymbol{U}-128*\mathbf{1}|_2|\boldsymbol{V}-128*\mathbf{1}|_2}$ This distance function is firstly shift each vector by $128 * \mathbf{1}$, then calculate their correlation.

	1	**3**	**5**	**7**	**9**	**11**	**13**	**15**	**17**	**19**
Training Accuracy(%)	0.70	0.71	0.73	0.74	0.74	0.74	0.75	0.75	0.75	0.75
Developing Accuracy(%)	0.80	0.81	0.82	0.82	0.81	0.81	0.81	0.81	0.81	0.81
Testing Accuracy(%)	0.80	0.79	0.81	0.81	0.81	0.80	0.80	0.80	0.80	0.80

Code : the code is run_1_KNN_interesting_2(K, set)

2 Linear Regression

2.1 Report

2.1.1 Question 12: Feature design

The following table show training and testing mean squared error using linear, linear + quadratic, and linear + quadratic + cubic features. We choose to use MSE instead of RSS because that

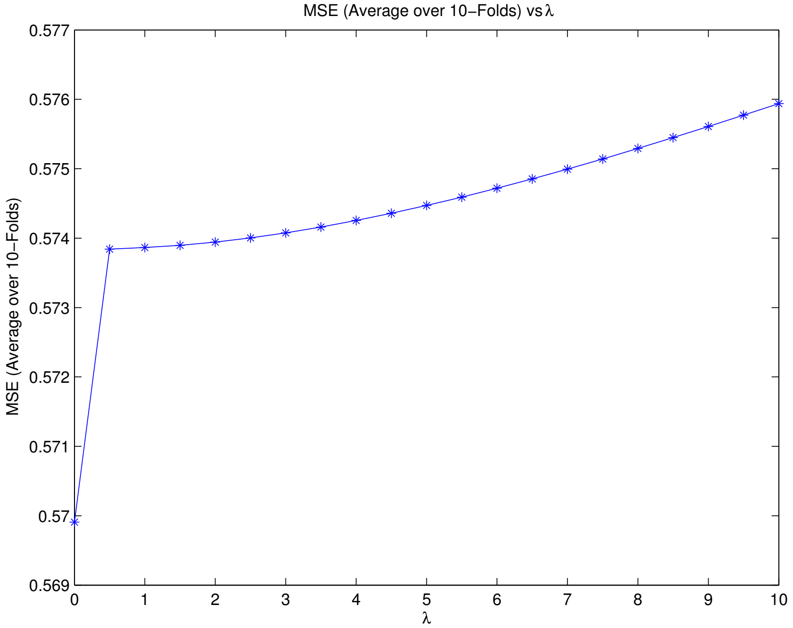

Figure 1: $||\boldsymbol{X}^t rain \boldsymbol{w}^\lambda - \boldsymbol{y}||_2^2$

Feature set	Training set MSE	Testing set MSE
Linear	0.5699	0.5739
Linear + quadratic	0.5124	0.5292
Linear + quadratic + cubic	0.4455	0.5026

Introducing linearly independent features help us to fit the training data better. However it might lead to overfitting, and thus increase the test set error. Fortunately this is not the case here. We can observe both training and testing mean squared error decrease as we add more features, but the improvement on the test data gets smaller as the order of extra features goes up. That means inevitably overfitting will occur and increase test set MSE. Note that across three feature sets, test error is always higher than training. This is the noise part caused by training/testing set mismatch.

2.1.2 Question 13: Regularization

Using 10-fold cross validation and linear features, we get the accuracy maximizing $\lambda \in [0 : 0.5 : 10]$ is 0. We think the reason is because, when we are using the linear features, we have only (11+1) features, and 3898 * 0.9 training data, the data size is big enough to help us get good w for this simple linear model. The result figure is figure 1,2,3. We separate the $||w^\lambda||_2^2$ into two figures, because when the λ change from 0 to 0.5, the $||w^\lambda||_2^2$ drop a lot. Two figures for this just for better showing the result.

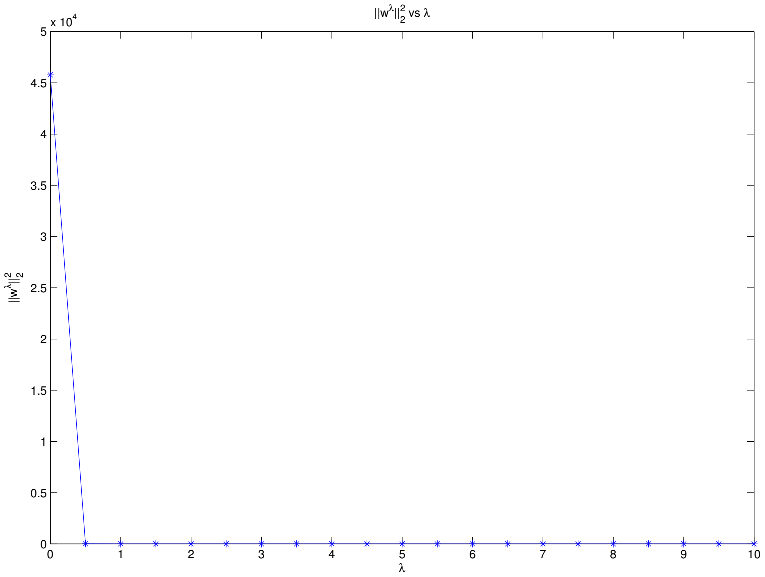

Figure 2: $||w^{\lambda}||_2^2$ vs λ

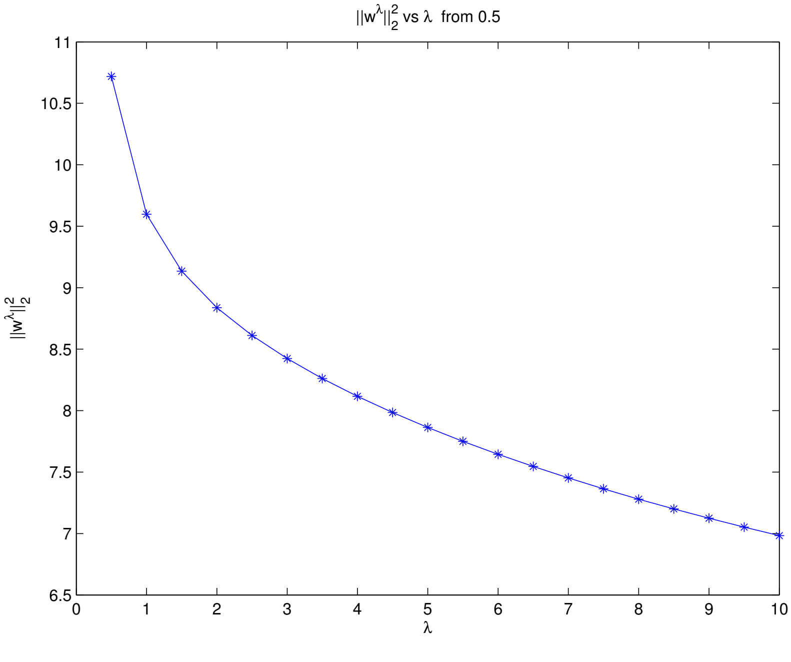

Figure 3: $||w^{\lambda}||_2^2$ vs λ from 0.5

2.1.3 Question 14: The role of the regularizer in the model

The regularizer can prevent overfitting to the training set. It also makes the model simpler by suppressing $|\boldsymbol{w}|_2^2$. When there is a big mismatch between training and test set, λ plays an important role on balancing the known versus unseen data. How to choose this parameter becomes a tough problem because even if you do cross validation, the mismatch always exists and therefore it is highly unlikely one can obtain an optimal λ that maximizes the accuracy on test set.

2.2 Code

We use Linear + quadratic + cubic features since this gives the best accuracy.