1. Bias Variance Trade-off

(a)

Given, L_2 regularized linear regression,

$$\hat{\beta}_\lambda = arg\,min_\beta \left\{ \frac{1}{n} \sum_{i=1}^{n} \left(y_i - x_i^T \beta\right)^2 + \lambda \|B\|_2^2 \right\}$$

The closed form solution for $\hat{\beta}_\lambda = (X^T X + \lambda I)^{-1} X^T y = (X^T X + \lambda I)^{-1} X^T (X\beta^* + \varepsilon)$

By using affine transformation, we get the distribution as

$$\boxed{\hat{\beta}_\lambda \sim N\left((X^T X + \lambda I)^{-1} X^T X \beta^*, (X^T X + \lambda I)^{-1} X^T X (X^T X + \lambda I)^{-1}\right)}$$

(b)

We know that $Bias = E\left[x^T \hat{\beta}_\lambda - x^T \beta^*\right] = x^T E\left[\hat{\beta}_\lambda - \beta^*\right]$

Using this in the equation derived in the first part, we get,

$$\boxed{x^T((X^T X + \lambda I)^{-1} X^T X - I)\beta^*)}$$

(c)

By using affine transformation we realize that $x^T E\left[\hat{\beta}_\lambda - E[\hat{\beta}_\lambda]\right]$ is a zero mean Gaussian random variable with variance

$$X^T (X^T X + \lambda I)^{-1} X^T X (X^T X + \lambda I)^{-1} x = \|X(X^T X + \lambda I)^{-1} x\|_2^2$$

$$\boxed{= \|X(X^T X + \lambda I)^{-1} x\|_2^2}$$

(d)

The error can be written as

$$Error = Bias^2 + Variance + Noise$$

$$\boxed{= x^T((X^T X + \lambda I)^{-1} X^T X - I)\beta^*) + \|X(X^T X + \lambda I)^{-1} x\|_2^2 + Noise}$$

2. Kernel Construction

(a) Given that the kernel are semidefinite functions, we can say that K_3 is also semidefinite as $K_3 = a_1K_1 + a_2K_2$. Since $a_1, a_2 > 0$ we can come to a conclusion at K_3 is positive semidefinite, hence a valid kernel function.

(b) Given, $K_4(x, x') = f(x)f(x')$, f(.) is a real valued function.

Consider, $K_4^T = [K_4(x_i, x_j)]^T = K_4(x_j, x_i) = f(x_j)(x_i)$. Now we know that f(x)f(x') is commutative so we can say $f(x_j)(x_i) = f(x_i)f(x_j)$ which is same as $K_4(x_i,x_j) = K_4$. Hence, we can say that it is symmetric.

Also, since $f(x_i) = f(x_j)$, we can write $v^T K_4 v = \left(\sum_i v_i f(x_i)\right)^2 \geq 0$ which means that K_4 is positive semidefinite, hence we can say that K_4 is a valid kernel function.

(c) The Schur theorem of products states that the Hadamard product of two positive semidefinite matrices results in a matrix which is semi definite as well. Since we've been given that $K_5(x,x') = K_1(x,x') \cdot K_2(x,x')$, we can say that K_5 is also semidefinite.

3. Kernel Regression

(a)

The given equation can be represented as,

$$= \min_w w^T X^T X w - 2y^T X w + \lambda w^T w$$

Differentiating w.r.t. to w and equating to 0 we get,

$$\frac{\delta L(w)}{\delta w} = 2X^T Xw - 2X^T y + 2\lambda w = 0$$

$$\boxed{w^* = (X^T X + \lambda I_D)^{-1} X^T y}$$

(b)

The closed form solution for the above equation can be written by using the identity on matrices given in the question, that is for any matrix $P \in R^{pxp}, B \in R^{qxp}, R \in R^{qxq}$ *and assuming that matrix inversion is valid, we get,* $(P^{-1} + B^T R^{-1} B)^{-1} B^T R^{-1} = PB^T (BPB^T + R)^{-1}$ with $R = I_N$ *and* $B = \phi$ *and* $P = \frac{I_D}{N}$, we get

$$\boxed{w^* = \phi^T (\phi\phi^T + \lambda I_N)^{-1} y}$$

(c)

The kernel matrix has been given as $K_{ij} = \Phi_i > \Phi_j$. Using this we classify w*,

$$\hat{y} = (\phi^T (\phi\phi^T + \lambda I_N)^{-1} y)^T \phi(x)$$

$$= y^T (\phi\phi^T + \lambda I_N)^{-1} \phi\phi(x)$$

Representing $\phi\phi^T$ as K

$$\boxed{\hat{y} = y^T (K + \lambda I_N)^{-1} K(x)}$$

(d) Complexities

Linear Ridge Regression:

1. $X^T X$ *is* $O(ND^2)$
2. $X^T X + \lambda I$ *is* $O(D^3)$
3. $X^T Y$ *is* $O(ND)$
4. $(X^T X)^{-1} X^T Y$ *is* $O(D^3)$
5. *Total complexity is* $O(ND^2 + D^3)$

Prediction is just printing out values and it can be done in O(D) time

Kernel Ridge Regression:

1. K(x,z) for N^2 entries of matrix K : $O(kN^2)$
2. $(K + \lambda I_N)^{-1}: O(N^3)$
3. $y^T(K + \lambda I_N)^{-1}: O(N^2)$
4. Total complexity is $O((k+N)N^2)$

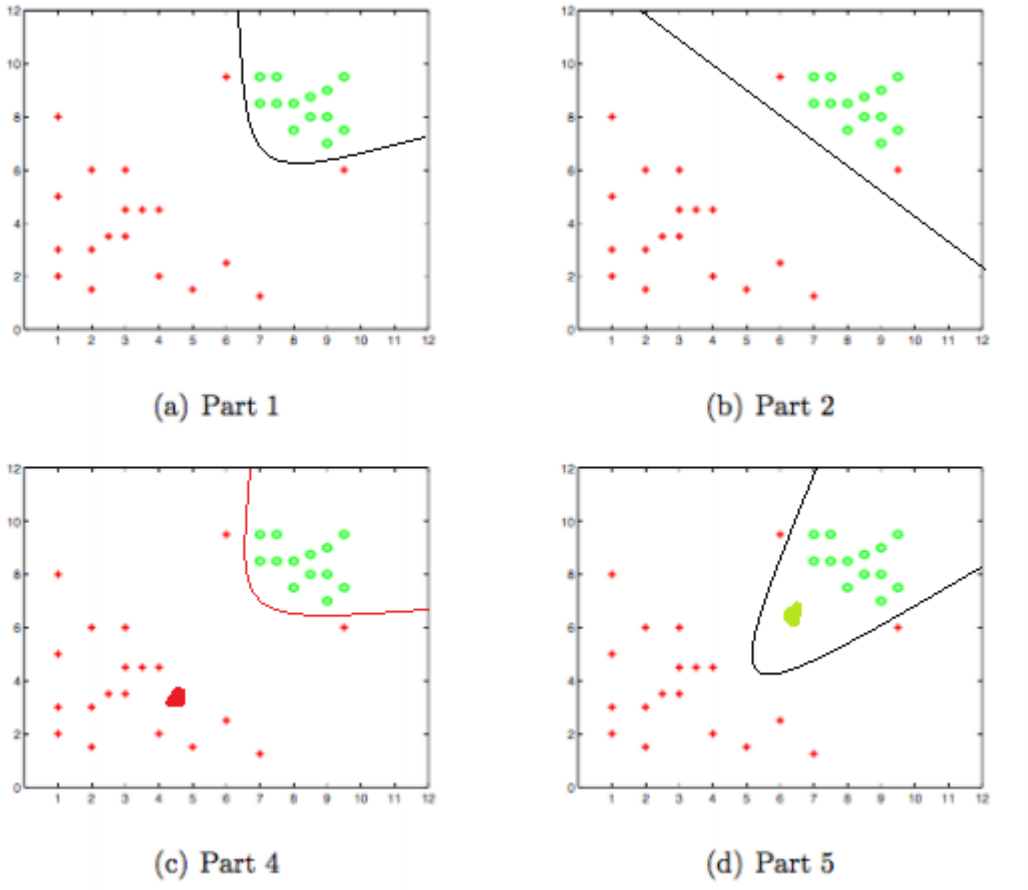
For prediction we need to compute $\hat{y} = MK(x),\ K(x)\ take\ O(kN)\ time$, total time is

$$O(\mathrm{kN} + \mathrm{N}) = \mathrm{O(N)}$$

4. Support Vector Machine

1. No, the positive examples aren't linearly separable from negative examples in original space
2. $w = (0,0,0,1)^T$
3. Consider the point (1.25,1.25) and assign a negative value in the positive space. It will be classified wrongly.
4. The kernel is a polynomial kernel of degree 2 because it has X_1 and X_2

5. SVMs and slack penalty C



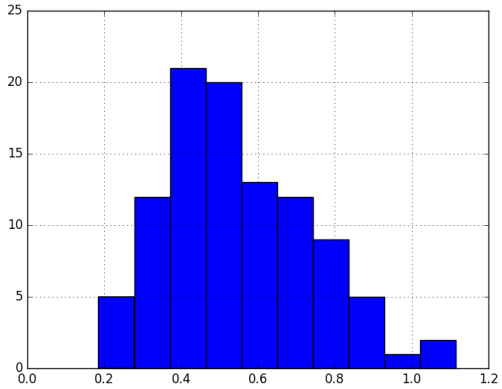
(a) Part 1 (b) Part 2

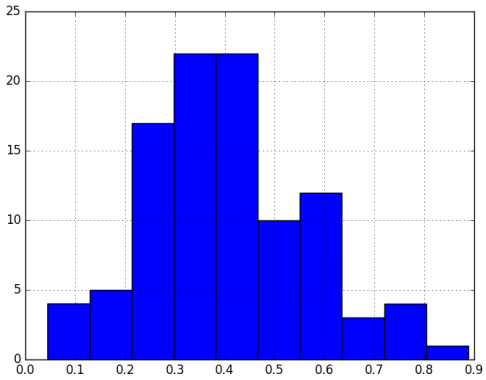
(c) Part 4 (d) Part 5

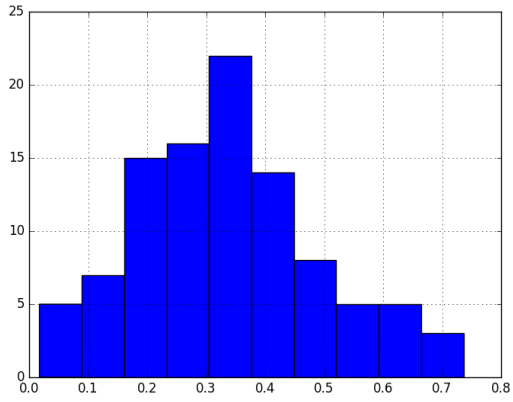
6. Programming

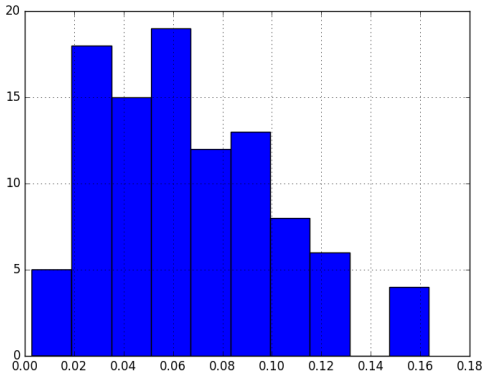
(a) Bias Variance Trade Off

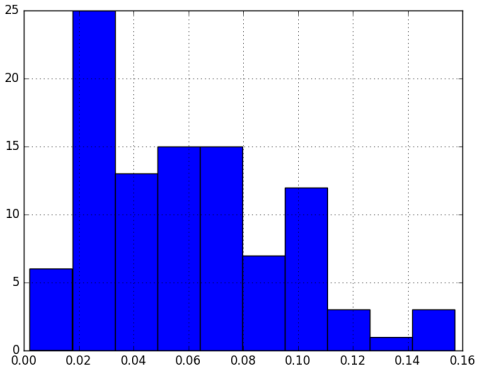
a) Histograms for Mean Square Error for g1 to g6 functions for sample size 10 using Linear Regression

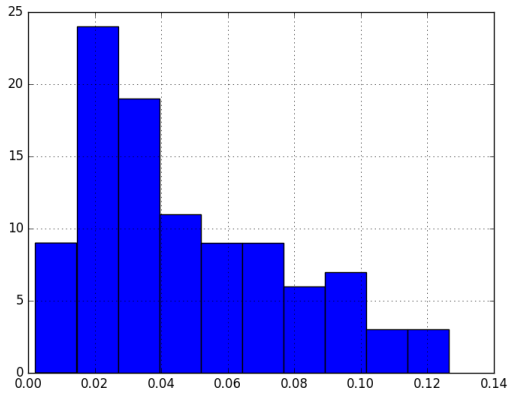












Bias^2

Bias^2 of g1 0.448585870268

Bias^2 of g2 0.366155378308

Bias^2 of g3 0.367521772501

Bias^2 of g4 0.239763976183

Bias^2 of g5 0.238790598054

Bias^2 of g6 0.236974744944

Variance

Variance of g1 0.0

Variance of g2 0.0403290322831

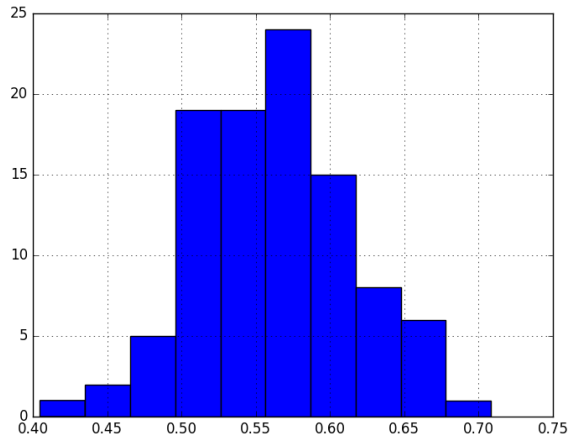
Variance of g3 0.0630415202832

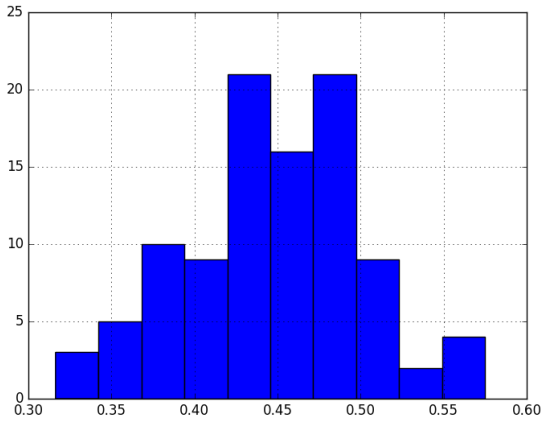
Variance of g4 0.0591856474388

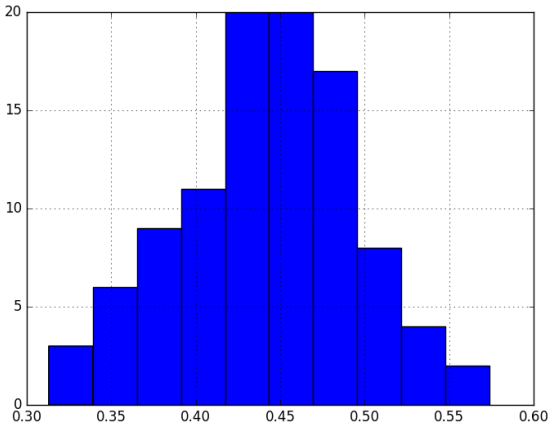
Variance of g5 0.0642077585068

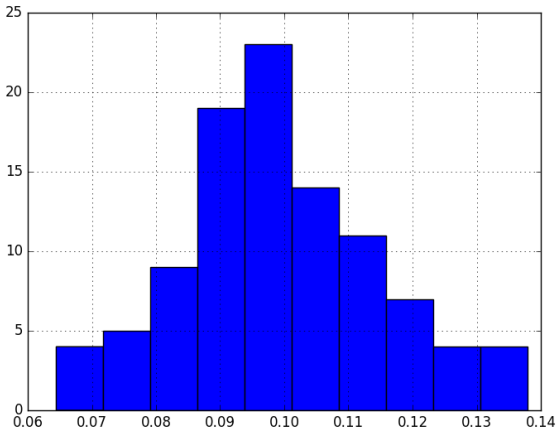
Variance of g6 0.104307510628

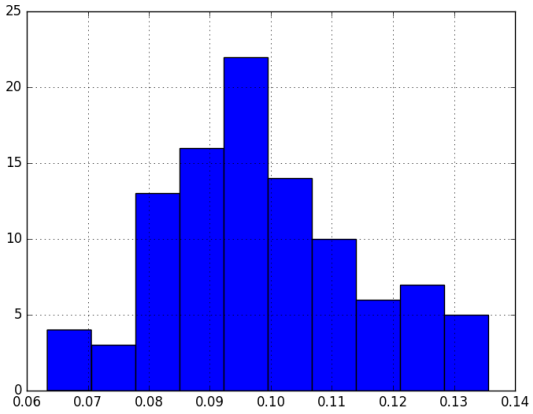
b) b) Histograms for Mean Square Error for g1 to g6 functions for sample size 100 using Linear Regression

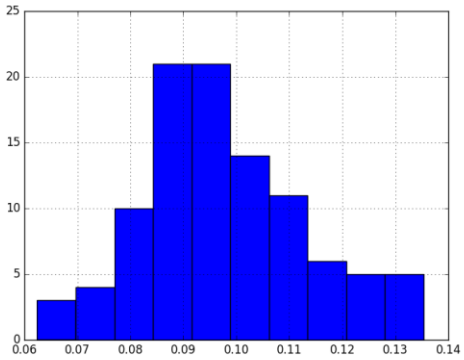





20
15
10
5
0
0.30
0.35
0.40
0.45
0.50
0.55
0.60


25
20
15
10
5
0
0.06
0.07
0.08
0.09
0.10
0.11
0.12
0.13
0.14


25
20
15
10
5
0
0.06
0.07
0.08
0.09
0.10
0.11
0.12
0.13
0.14



Bias^2

Bias^2 of g1 0.463242956313

Bias^2 of g2 0.364743010872

Bias^2 of g3 0.364814208367

Bias^2 of g4 0.238962270444

Bias^2 of g5 0.238936452815

Bias^2 of g6 0.23871713744

Variance

Variance of g1 0.0

Variance of g2 0.00434089571324

Variance of g3 0.00579664984853

Variance of g4 0.00595899502307

Variance of g5 0.00632492431404

Variance of g6 0.00665534004691

c)

From the above results generated by running Linear Regression on different sample sizes (10 and 100) we see that the variance drops a lot (almost by a factor of 10) but the bias seems to remain the same, but on careful evaluation we see that the bias has increased by a minute factor as well. By varying the model complexity we see that the bias and variance drop to a certain extent to a minimum value and then start increasing steadily (overfitting). The same trend can be seen for model complexity changes when the sample size is increased to 100 from 10.

d) Ridge Regression for sample size 100

Ridge Regression for Sample Size 100 and lambda value 0.001

Bias^2 of g4 0.240508277405

Variance of g4 0.00435662017576

Ridge Regression for Sample Size 100 and lambda value 0.003

Bias^2 of g4 0.238208596784

Variance of g4 0.00426454051689

Ridge Regression for Sample Size 100 and lambda value 0.01

Bias^2 of g4 0.231353142552

Variance of g4 0.00593173282375

Ridge Regression for Sample Size 100 and lambda value 0.03

Bias^2 of g4 0.230282832607

Variance of g4 0.00509836906145

Ridge Regression for Sample Size 100 and lambda value 0.1

Bias^2 of g4 0.237245053816

Variance of g4 0.00541779279626

Ridge Regression for Sample Size 100 and lambda value 0.3

Bias^2 of g4 0.238166858801

Variance of g4 0.00556519536282

Ridge Regression for Sample Size 100 and lambda value 1.0

Bias^2 of g4 0.233557043403

Variance of g4 0.00541523390533

So, when we run Ridge Regression on the dataset with sample size 100 for lambda values varying from 0.001 to 1.0 we get almost constant bias values but on closer inspection we see that there is a slight increase in the values and the variance decreases till the lambda value 0.003 and then fluctuates a bit after which it goes on to increase slightly for the increase in lambda values.

(b) Linear and Kernel SVM

Linear SVM

c = 0.000244140625

Cross Validation Accuracy = 55.75%

c = 0.0009765625

Cross Validation Accuracy = 88.65%

c = 0.00390625

Cross Validation Accuracy = 91.25%

c = 0.015625

Cross Validation Accuracy = 92.9%

c = 0.0625

Cross Validation Accuracy = 94.35%

c = 0.25

Cross Validation Accuracy = 94.35%

c = 1

Cross Validation Accuracy = 94.55%

c = 4

Cross Validation Accuracy = 94.95%

c = 16

Cross Validation Accuracy = 94.25%

The average training time for Linear SVM is 0.137489627909 seconds

---

a) Polynomial Kernel

c = 0.015625, degree = 1

Cross Validation Accuracy = 55.75%

c = 0.0625, degree = 1

Cross Validation Accuracy = 89.75%

c = 0.25, degree = 1

Cross Validation Accuracy = 91.25%

c = 1, degree = 1

Cross Validation Accuracy = 93.35%

c = 4, degree = 1

Cross Validation Accuracy = 94.5%

c = 16, degree = 1

Cross Validation Accuracy = 94.65%

c = 64, degree = 1

Cross Validation Accuracy = 94.6%

c = 256, degree = 1

Cross Validation Accuracy = 94.55%

c = 1024, degree = 1

Cross Validation Accuracy = 94.4%

c = 4096, degree = 1

Cross Validation Accuracy = 94.55%

c = 16384, degree = 1

Cross Validation Accuracy = 94.25%

c = 0.015625, degree = 2

Cross Validation Accuracy = 55.75%

c = 0.0625, degree = 2

Cross Validation Accuracy = 88.75%

c = 0.25, degree = 2

Cross Validation Accuracy = 91.7%

c = 1, degree = 2

Cross Validation Accuracy = 93.25%

c = 4, degree = 2

Cross Validation Accuracy = 95.05%

c = 16, degree = 2

Cross Validation Accuracy = 95.35%

c = 64, degree = 2

Cross Validation Accuracy = 95.75%

c = 256, degree = 2

Cross Validation Accuracy = 96.15%

c = 1024, degree = 2

Cross Validation Accuracy = 96.5%

c = 4096, degree = 2

Cross Validation Accuracy = 96.55%

c = 16384, degree = 2

Cross Validation Accuracy = 96.05%

c = 0.015625, degree = 3

Cross Validation Accuracy = 55.75%

c = 0.0625, degree = 3

Cross Validation Accuracy = 72.05%

c = 0.25, degree = 3

Cross Validation Accuracy = 91.85%

c = 1, degree = 3

Cross Validation Accuracy = 92.9%

c = 4, degree = 3

Cross Validation Accuracy = 94.9%

c = 16, degree = 3

Cross Validation Accuracy = 95.85%

c = 64, degree = 3

Cross Validation Accuracy = 96.5%

c = 256, degree = 3

Cross Validation Accuracy = 97.1%

c = 1024, degree = 3

Cross Validation Accuracy = 96.9%

c = 4096, degree = 3

Cross Validation Accuracy = 96.6%

c = 16384, degree = 3

Cross Validation Accuracy = 96.55%

Polynomial Kernel MAX is 97.1 at c 4^4.0 and degree 3

The average training time for Polynomial Kernel is 0.193217585785 seconds

---

b) RBF Kernel

c = 0.015625, gamma = 6.103515625e-05

Cross Validation Accuracy = 55.75%

c = 0.0625, gamma = 6.103515625e-05

Cross Validation Accuracy = 55.75%

c = 0.25, gamma = 6.103515625e-05

Cross Validation Accuracy = 55.75%

c = 1, gamma = 6.103515625e-05

Cross Validation Accuracy = 55.75%

c = 4, gamma = 6.103515625e-05

Cross Validation Accuracy = 67.65%

c = 16, gamma = 6.103515625e-05

Cross Validation Accuracy = 90.7%

c = 64, gamma = 6.103515625e-05

Cross Validation Accuracy = 91%

c = 256, gamma = 6.103515625e-05

Cross Validation Accuracy = 93.4%

c = 1024, gamma = 6.103515625e-05

Cross Validation Accuracy = 94.75%

c = 4096, gamma = 6.103515625e-05

Cross Validation Accuracy = 94.6%

c = 16384, gamma = 6.103515625e-05

Cross Validation Accuracy = 94.4%

c = 0.015625, gamma = 0.000244140625

Cross Validation Accuracy = 55.75%

c = 0.0625, gamma = 0.000244140625

Cross Validation Accuracy = 55.75%

c = 0.25, gamma = 0.000244140625

Cross Validation Accuracy = 55.75%

c = 1, gamma = 0.000244140625

Cross Validation Accuracy = 67.8%

c = 4, gamma = 0.000244140625

Cross Validation Accuracy = 90.7%

c = 16, gamma = 0.000244140625

Cross Validation Accuracy = 91.25%

c = 64, gamma = 0.000244140625

Cross Validation Accuracy = 93.45%

c = 256, gamma = 0.000244140625

Cross Validation Accuracy = 94.5%

c = 1024, gamma = 0.000244140625

Cross Validation Accuracy = 94.05%

c = 4096, gamma = 0.000244140625

Cross Validation Accuracy = 94.15%

c = 16384, gamma = 0.000244140625

Cross Validation Accuracy = 94.85%

c = 0.015625, gamma = 0.0009765625

Cross Validation Accuracy = 55.75%

c = 0.0625, gamma = 0.0009765625

Cross Validation Accuracy = 55.75%

c = 0.25, gamma = 0.0009765625

Cross Validation Accuracy = 66.6%

c = 1, gamma = 0.0009765625

Cross Validation Accuracy = 90.8%

c = 4, gamma = 0.0009765625

Cross Validation Accuracy = 91.35%

c = 16, gamma = 0.0009765625

Cross Validation Accuracy = 93.35%

c = 64, gamma = 0.0009765625

Cross Validation Accuracy = 94.25%

c = 256, gamma = 0.0009765625

Cross Validation Accuracy = 94.1%

c = 1024, gamma = 0.0009765625

Cross Validation Accuracy = 94.65%

c = 4096, gamma = 0.0009765625

Cross Validation Accuracy = 95.65%

c = 16384, gamma = 0.0009765625

Cross Validation Accuracy = 96.6%

c = 0.015625, gamma = 0.00390625

Cross Validation Accuracy = 55.75%

c = 0.0625, gamma = 0.00390625

Cross Validation Accuracy = 63.65%

c = 0.25, gamma = 0.00390625

Cross Validation Accuracy = 90.6%

c = 1, gamma = 0.00390625

Cross Validation Accuracy = 91.3%

c = 4, gamma = 0.00390625

Cross Validation Accuracy = 93.45%

c = 16, gamma = 0.00390625

Cross Validation Accuracy = 94.6%

c = 64, gamma = 0.00390625

Cross Validation Accuracy = 95.45%

c = 256, gamma = 0.00390625

Cross Validation Accuracy = 95.05%

c = 1024, gamma = 0.00390625

Cross Validation Accuracy = 96.9%

c = 4096, gamma = 0.00390625

Cross Validation Accuracy = 96.6%

c = 16384, gamma = 0.00390625

Cross Validation Accuracy = 95.8%

c = 0.015625, gamma = 0.015625

Cross Validation Accuracy = 56.15%

c = 0.0625, gamma = 0.015625

Cross Validation Accuracy = 90.4%

c = 0.25, gamma = 0.015625

Cross Validation Accuracy = 91.45%

c = 1, gamma = 0.015625

Cross Validation Accuracy = 93.5%

c = 4, gamma = 0.015625

Cross Validation Accuracy = 94.9%

c = 16, gamma = 0.015625

Cross Validation Accuracy = 95.5%

c = 64, gamma = 0.015625

Cross Validation Accuracy = 96.9%

c = 256, gamma = 0.015625

Cross Validation Accuracy = 97.1%

c = 1024, gamma = 0.015625

Cross Validation Accuracy = 96.15%

c = 4096, gamma = 0.015625

Cross Validation Accuracy = 96.6%

c = 16384, gamma = 0.015625

Cross Validation Accuracy = 96.45%

c = 0.015625, gamma = 0.0625

Cross Validation Accuracy = 87.45%

c = 0.0625, gamma = 0.0625

Cross Validation Accuracy = 91.8%

c = 0.25, gamma = 0.0625

Cross Validation Accuracy = 93.65%

c = 1, gamma = 0.0625

Cross Validation Accuracy = 95.45%

c = 4, gamma = 0.0625

Cross Validation Accuracy = 96.6%

c = 16, gamma = 0.0625

Cross Validation Accuracy = 97.35%

c = 64, gamma = 0.0625

Cross Validation Accuracy = 96.15%

c = 256, gamma = 0.0625

Cross Validation Accuracy = 96.6%

c = 1024, gamma = 0.0625

Cross Validation Accuracy = 96.2%

c = 4096, gamma = 0.0625

Cross Validation Accuracy = 96.45%

c = 16384, gamma = 0.0625

Cross Validation Accuracy = 96.7%

c = 0.015625, gamma = 0.25

Cross Validation Accuracy = 60.4%

c = 0.0625, gamma = 0.25

Cross Validation Accuracy = 92.2%

c = 0.25, gamma = 0.25

Cross Validation Accuracy = 96%

c = 1, gamma = 0.25

Cross Validation Accuracy = 97.5%

c = 4, gamma = 0.25

Cross Validation Accuracy = 96.85%

c = 16, gamma = 0.25

Cross Validation Accuracy = 97.45%

c = 64, gamma = 0.25

Cross Validation Accuracy = 96.5%

c = 256, gamma = 0.25

Cross Validation Accuracy = 96.85%

c = 1024, gamma = 0.25

Cross Validation Accuracy = 97.1%

c = 4096, gamma = 0.25

Cross Validation Accuracy = 96.8%

c = 16384, gamma = 0.25

Cross Validation Accuracy = 97.4%

RBF Kernel MAX is 97.5 at c 4^0.0 and gamma 4^-1.0

The average training time for Polynomial Kernel is 0.158710207258 seconds

c) libsvm.py

Kernel Selected based on accuracy: RBF Kernel with c 4^0.0 and gamma 4^-1.0

Accuracy = 95.55% (1911/2000) (classification)

COLLABORATION

Brain stormed and Collaborated with Adarsha Desai and Ravishankar Sivaraman for this assignment.