1. Bias Variance Trade-of

(a)

We are given the linear model,

$$y = x^T\beta^* + \varepsilon$$

And L_2 linear regression with the regularization parameter $\lambda \geq 0$

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

The closed form solution for $\hat{\beta}_\lambda$ is

$$\hat{\beta}_\lambda \;=\; (X^TX \;+\; \lambda I)^{-1}\, X^Ty$$

$$\hat{\beta}_\lambda \;=\; (X^TX \;+\; \lambda I)^{-1}\, X^T(X\beta^* + \varepsilon)$$

Using the affine transformation to a Gaussian distribution, we get

$$\hat{\beta}_\lambda \;\sim\; N(\,(X^TX \;+\; \lambda I)^{-1}\, X^TX\beta^*, (X^TX \;+\; \lambda I)^{-1}X^TX\,(X^TX \;+\; \lambda I)^{-1})$$

1. (b)

We know,

$$Bias \;=\; E\left[x^T\,\hat{\beta}_\lambda \;- x^T\,\beta^*\right] \;= x^T\,E[\beta_\lambda \;-\; \beta^*]$$

Substituting for β_λ from the above,

$$Bias \;=\; [(X^TX \;+\; \lambda I)^{-1}X^TX - I\;]\;\beta^*$$

1. (c)

When computing the variance, if we use the affine transformation of Gaussian distribution, we realize that we get a zero mean Gaussian random variable. But we know that the square of a Guassian variable is a χ^2 random variable which we can use to get.

$$E\;[\;(x^T\left(\hat{\beta}_\lambda \;-\; E\left[\hat{\beta}_\lambda\right]\right)^2] \;= \|X(X^TX \;+\; \lambda I)^{-1}\,X\|_2^2$$

1. (d)

The bias-variance trade-off for this can be written as

$$E\;\left[\;\left(x^T\hat{\beta}_\lambda - \;x^T\,\beta^*\right)^2\right] = \;[X^T((X^TX \;+\; \lambda I)^{-1}X^TX - I)\beta^* + \|X(X^TX \;+\; \lambda I)^{-1}\,X\|_2^2 + const.$$

As λ increases, bias increases and variance decreases

2. Kernel Construction

(a)

We know that K_1 and K_2 are symmetric kernel matrices.
And,

$$K_3 = a_1 K_1 + a_2 K_2$$

.
Therefore K_3 will also be a symmetric.
Now, $a_1, a_2 \geq 0$ and K_1 and K_2 are positive semi-definite
Hence,

$$v^T K_3 v = a_1 v^T K_1 v + a_2 v^T K_2 v$$

will also be ≥ 0
This implies that K_3 is also positive semi-definite and is a valid kernel matrix

2. (b)

For the kernel

$K_4(x, x') = f(x) f(x')$

Now,

$K_4^T = [K_4(x, x')]^T = [K_4(x', x)] = f(x') f(x) = f(x) f(x') = [K_4(x, x')] = K_4$

Therefore, K_4 is symmetric

$$v^T K_4 v = \sum_i \sum_j v_i f(x_i) f(x_j) v_j$$

Separating the summations

$$v^T K_4 v = \sum_i v_i f(x_i) \sum_j v_j f(x_j) = \left(\sum_i v_i f(i) \right)^2$$

Which is always ≥ 0
Hence, K_4 is positive semi-definite

2. (c)

We have
$K_5(x, x') = K_1(x, x')\, K_2(x, x')$

Now, Schur product theorem states that the Hadamard product of two positive semidefinite matrices is also positive semi definite.

Hence K_5 is positive semidefinite and hence a valid kernel function

Alternatively,

We have $k_5(x, x')$ as

$$k_5(x, x') = k_1(x, x')k_2(x, x') = \phi_1(x)^T \phi_1(x')\phi_2(x)^T \phi_2(x')$$

Substituting for $\phi_1 and\ \phi_2$

$$\begin{aligned} k_5(x, x') &= \sum_{i=1}^{T} \phi_{1i}(x)\phi_{1i}(x') \sum_{j=1}^{T} \phi_{2j}(x)\phi_{2j}(x') \\ &= \sum_{i=1}^{T} \sum_{j=1}^{T} \phi_{1i}(x)\phi_{1i}(x')\phi_{2j}(x)\phi_{2j}(x') \\ &= \sum_{i=1}^{T} \sum_{j=1}^{T} (\phi_{1i}(x)\phi_{2j}(x))(\phi_{1i}(x')\phi_{2j}(x')) \\ &= \sum_{k=1}^{T^2} \phi_{5k}(x)\phi_{5k}(x') \\ &= \phi_5(x)^T \phi_5(x') \end{aligned}$$

With $\phi_5(x)$ being a feature transformation of length T^2. Since $\phi_5(x)$ is valid, K_5 is a valid kernel

3. Kernel Regression

(a)

We know,

$$\min_w \mathcal{L}(w) = \min_w (y - Xw)^T (y - Xw) + \lambda w^T w$$

$$\min_w \mathcal{L}(w) = \min_w w^T X^T Xw - 2y^T Xw + \lambda w^T w$$

Taking the gradient to minimize this,

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

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

3. (b)

The residual sum of squares for this can be written as

$$RSS(w) = \min_{w} \sum_{n} \left(y_i - w^T \Phi(x_i)\right)^2 + k\|w\|^2$$

Taking the gradient,

$$\nabla_w RSS(w) = 2 \sum_{\mathrm{n}} \left(y_i - w^T \Phi(x_i)\right)(-\Phi(x_i)) + 2\mathrm{kw} = 0$$

The design matrix from this equates to,

$$\Phi^T Y (\Phi\Phi^T + \lambda I_N) w$$

$$w = (\Phi\Phi^T + \lambda I_N)^{-1} \Phi^T Y$$

Applying Inverse Matrix lemma, we get,

$$w = \Phi^T (\Phi\Phi^T + \lambda I_N)^{-1} Y$$

3. (c) To classify with w^* , we compute:

$$\hat{y} = w^{*T} \varphi(x) = (\Phi^T (\Phi\Phi^T + \lambda I_N)^{-1} y)^T \varphi(x)$$

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

Or,

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

Where $\Phi\Phi^T$ is K and $\Phi \varphi(x)$ is $\kappa(x)$

3. (d)

Linear regression training complexity:

Training for linear ridge regression involves computing $w^* = (X^T X + \lambda I_D)^{-1} X^T y$

The computation complexities for different operations are:

$X^T X$: $O(ND^2)$

Add $X^T X$ and λI_D: $O(D^2)$

$X^T y$: $O(ND)$

Inverting$(X^T X + \lambda I_D)$ and multiplying with$X^T y$: $\mathrm{O}(\mathrm{D}^3)$

Total: $O(ND^2 + D^3)$.

Prediction Complexity of Linear Ridge Regression: $O(D)$

Kernel Ridge regression training complexity:

Computing Kernel Matrix (K): Requires computing $\Phi\Phi^T$

Inverting$(K + \lambda I_N)$: $O(N^3)$

multiplying $y^T and\ (K\ +\ \lambda I_N)^{-1}$: $\mathrm{O}(\mathrm{N}^3)$

Total: $O(kN^2\ +\ N^3\ +\ N^2\ )\ =\ O((k\ +\ N)N^2\ )$.

4. Support Vector Machine

(a)

This is similar to the XOR problem. The positive examples are not linearly separable from the negative examples for this.

4. (b)

New feature space is $[1, x_1, x_2\ , x_1 x_2\ ]$
The points in the new feature space are (1,1,1,1), (1,-1,-1,1), (1,1,-1,-1) and (1,-1,1,-1)
The $x_1 x_2$ feature separates the classes with maximum margin
Hence, $\mathrm{w} =\ (0,0,0,1)^{\mathrm{T}}$

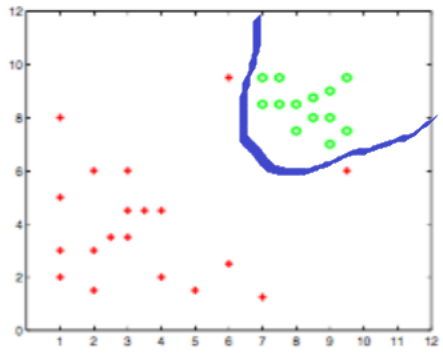
4. (c)

If we introduce a point (-0.5, 0.5) that belongs to positive class, then new feature space will classify it as negative which will be incorrect.

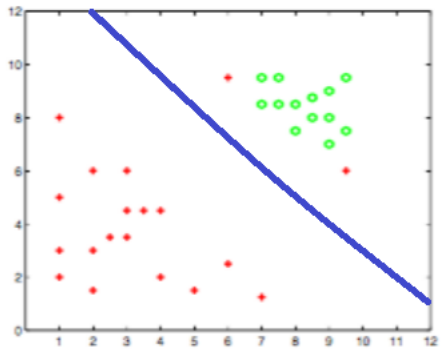
4. (d)

Kernel $K(x, x)$ will be a polynomial kernel with with degree 2 of the form $1 + X_1 X_1' + X_2 X_2' +\ X_1 X_1'\ X_2 X_2'$

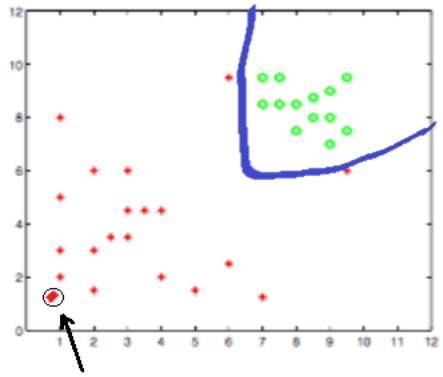
5. **SVMs and the slack penalty C**



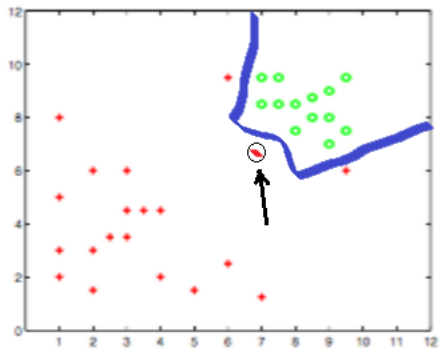
(a) Part 1



(b) Part 2



(c) Part 4



(d) Part 5

(a) For high values of slack penalty C, SVM will try and fit all the data with the decision boundary with very high precision.

(b) For low values of slack penalty C, SVM will under fit and mark the decision boundary with low precision.

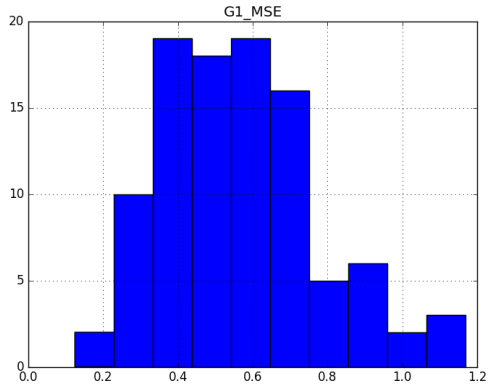
(c) Since the data is known to be error prone, we don't want all the points to be classified perfectly. Some of the outliers could actually be an effect of noise. So we prefer the solution with low slack penalties, because it maximizes the margin between the dominant sets of points.

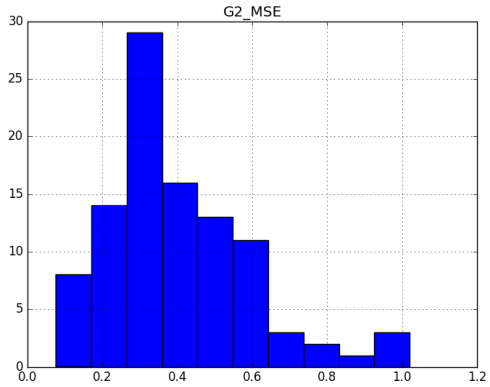
(d) A point as shown in the above figure (Part 4) which is present with other points of the same class and is far from the decision boundary will not affect the decision boundary.

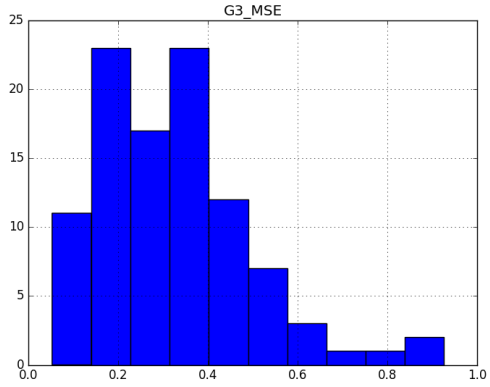
(e) For higher slack penalties, the model tends to overfit the data. Hence a data point in wrong region as in figure (part5) will change the boundary as the model will try and classify the point correctly.

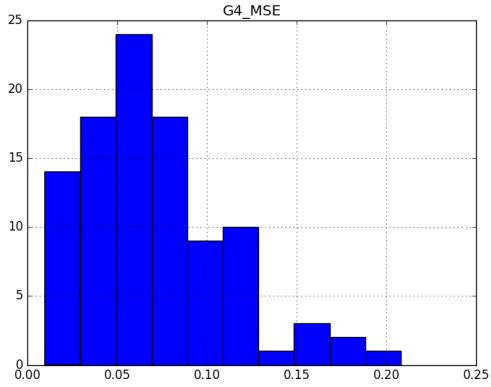
6. Programming`
Bias Variance Trade-of

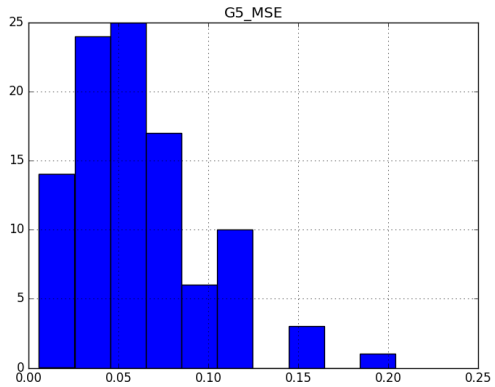
Histogram for N = 10 samples

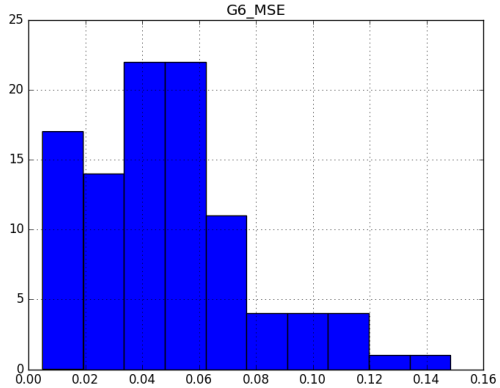




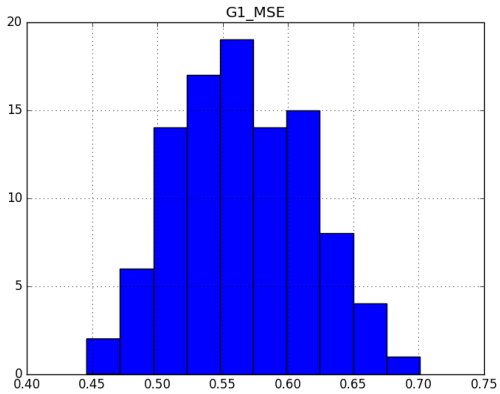


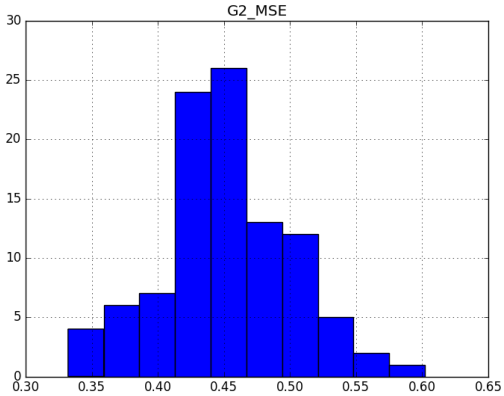


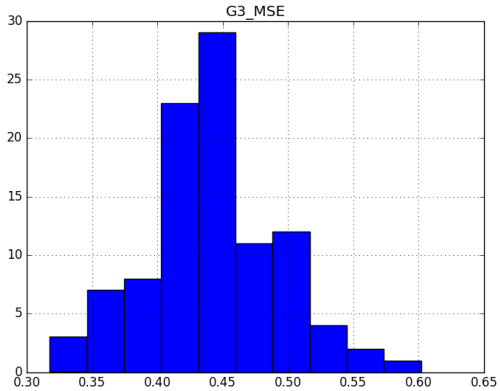


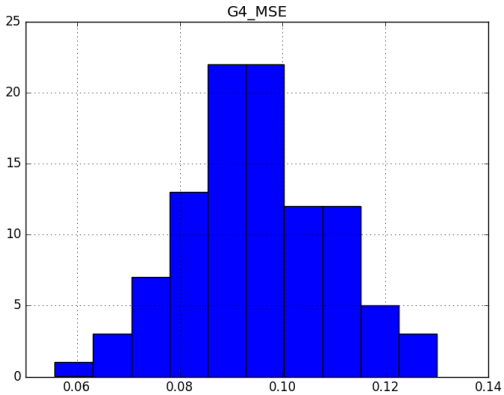


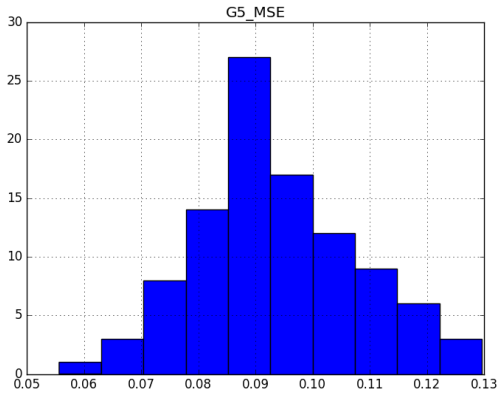
Histogram for N = 100 samples

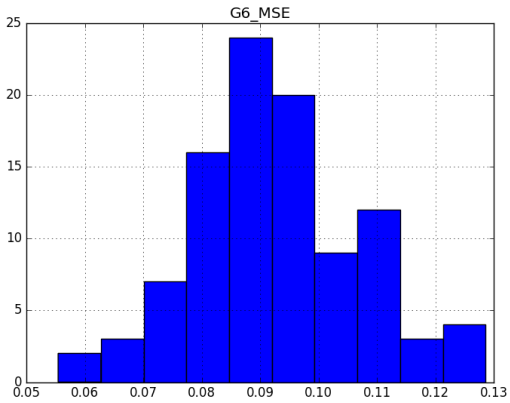












(a)

Linear Regression with 10 samples
BiasSquare for g1 = 0.462445048421
BiasSquare for g2 = 0.352559600884
BiasSquare for g3 = 0.352202439249
BiasSquare for g4 = 0.236183514097
BiasSquare for g5 = 0.237584735598
BiasSquare for g6 = 0.232118391919
Variance for g1 = 0.0
Variance for g2 = 0.0349481026495
Variance for g3 = 0.0574779379044
Variance for g4 = 0.0513013300539
Variance for g5 = 0.0786527656691
Variance for g6 = 0.233101059052

(b)

Linear Regression with 100 samples

BiasSquare for g1 = 0.455003907944
BiasSquare for g2 = 0.361076126215
BiasSquare for g3 = 0.361286620151
BiasSquare for g4 = 0.232677855623
BiasSquare for g5 = 0.232664458242
BiasSquare for g6 = 0.232339206998
Variance for g1 = 0.0
Variance for g2 = 0.00379198921376
Variance for g3 = 0.00571898666289
Variance for g4 = 0.00485452596204
Variance for g5 = 0.00511374138372
Variance for g6 = 0.00556450572967

(c)

Model Complexity : G4 has the best performance for both the bias and Variance. As the model complexity increases it is expected to perform better and then when it starts over fitting, the results deteriorate again. But since the dataset is small, we do not see this expected behaviour. The expected behaviour is seen for bigger dataset.

Effect of sample size:With increasing sample size, bias will increase and variance will decrease for higher order polynomial and bigger datasets. However, this is not evident here because for 100 datasets with 10 samples each, the dataset is small. For higher value samples such as 500, 1000, expected behaviour is noticed. As sample size increases bias variance tradeoff is more evident

(d)

Ridge Regression with 100 samples and Lambda = 0.001

BiasSquare for g4 = 0.23673874532
Variance for g4 = 0.00521786840081

Ridge Regression with 100 samples and Lambda = 0.003

BiasSquare for g4 = 0.23736907449
Variance for g4 = 0.00622274896596

Ridge Regression with 100 samples and Lambda = 0.01

BiasSquare for g4 = 0.237342794391
Variance for g4 = 0.00576478905356

Ridge Regression with 100 samples and Lambda = 0.03

BiasSquare for g4 = 0.231652937119
Variance for g4 = 0.00660376057348

Ridge Regression with 100 samples and Lambda = 0.1

BiasSquare for g4 = 0.240352593284
Variance for g4 = 0.00588495346783

Ridge Regression with 100 samples and Lambda = 0.3

BiasSquare for g4 = 0.236489176391
Variance for g4 = 0.00562530300673

Ridge Regression with 100 samples and Lambda = 1.0

BiasSquare for g4 = 0.235260950088
Variance for g4 = 0.00579897077342

Conclusion:

As lambda increases bias increases and variance decreases. But this trend cannot entirely be seen in this case as the data set is quite small.

7.

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 = 92.55%

C = 0.0625

Cross Validation Accuracy = 94.25%

C = 0.25

Cross Validation Accuracy = 94.7%

C = 1

Cross Validation Accuracy = 94.3%

C = 4

Cross Validation Accuracy = 93.95%

C = 16

Cross Validation Accuracy = 94.3%

Average time = 0.20161359954

Best : C = 0.25 , Degree = 3, Accuracy = 94.7%

Polynomial Kernel SVM

C = 0.015625 and Degree = 1

Cross Validation Accuracy = 55.75%

C = 0.015625 and Degree = 2

Cross Validation Accuracy = 55.75%

C = 0.015625 and Degree = 3

Cross Validation Accuracy = 55.75%

C = 0.0625 and Degree = 1

Cross Validation Accuracy = 90.15%

C = 0.0625 and Degree = 2

Cross Validation Accuracy = 88.5%

C = 0.0625 and Degree = 3

Cross Validation Accuracy = 71.7%

C = 0.25 and Degree = 1

Cross Validation Accuracy = 91.05%

C = 0.25 and Degree = 2

Cross Validation Accuracy = 91.8%

C = 0.25 and Degree = 3

Cross Validation Accuracy = 91.55%

C = 1 and Degree = 1

Cross Validation Accuracy = 92.95%

C = 1 and Degree = 2

Cross Validation Accuracy = 93%

C = 1 and Degree = 3

Cross Validation Accuracy = 92.85%

C = 4 and Degree = 1

Cross Validation Accuracy = 94.25%

C = 4 and Degree = 2

Cross Validation Accuracy = 94.3%

C = 4 and Degree = 3

Cross Validation Accuracy = 94.7%

C = 16 and Degree = 1

Cross Validation Accuracy = 94.6%

C = 16 and Degree = 2

Cross Validation Accuracy = 95.3%

C = 16 and Degree = 3

Cross Validation Accuracy = 96.2%

C = 64 and Degree = 1

Cross Validation Accuracy = 94.1%

C = 64 and Degree = 2

Cross Validation Accuracy = 96.05%

C = 64 and Degree = 3

Cross Validation Accuracy = 96.8%

C = 256 and Degree = 1

Cross Validation Accuracy = 93.45%

C = 256 and Degree = 2

Cross Validation Accuracy = 96%

C = 256 and Degree = 3

Cross Validation Accuracy = 96.15%

C = 1024 and Degree = 1

Cross Validation Accuracy = 94.75%

C = 1024 and Degree = 2

Cross Validation Accuracy = 96.45%

C = 1024 and Degree = 3

Cross Validation Accuracy = 96.7%

C = 4096 and Degree = 1

Cross Validation Accuracy = 94.1%

C = 4096 and Degree = 2

Cross Validation Accuracy = 96.65%

C = 4096 and Degree = 3

Cross Validation Accuracy = 96.75%

C = 16384 and Degree = 1

Cross Validation Accuracy = 94.5%

C = 16384 and Degree = 2

Cross Validation Accuracy = 96.2%

C = 16384 and Degree = 3

Cross Validation Accuracy = 96.25%

Average time = 0.264274037813

Best : C = 64, Degree = 3, Accuracy = 96.8%

RBF kernel SVM

C = 0.015625 and Gamma = 6.103515625e-05

Cross Validation Accuracy = 55.75%

C = 0.015625 and Gamma = 0.000244140625

Cross Validation Accuracy = 55.75%

C = 0.015625 and Gamma = 0.0009765625

Cross Validation Accuracy = 55.75%

C = 0.015625 and Gamma = 0.00390625

Cross Validation Accuracy = 55.75%

C = 0.015625 and Gamma = 0.015625

Cross Validation Accuracy = 55.8%

C = 0.015625 and Gamma = 0.0625

Cross Validation Accuracy = 87.5%

C = 0.015625 and Gamma = 0.25

Cross Validation Accuracy = 60.3%

C = 0.0625 and Gamma = 6.103515625e-05

Cross Validation Accuracy = 55.75%

C = 0.0625 and Gamma = 0.000244140625

Cross Validation Accuracy = 55.75%

C = 0.0625 and Gamma = 0.0009765625

Cross Validation Accuracy = 55.75%

C = 0.0625 and Gamma = 0.00390625

Cross Validation Accuracy = 64.3%

C = 0.0625 and Gamma = 0.015625

Cross Validation Accuracy = 90.3%

C = 0.0625 and Gamma = 0.0625

Cross Validation Accuracy = 92.05%

C = 0.0625 and Gamma = 0.25

Cross Validation Accuracy = 92.55%

C = 0.25 and Gamma = 6.103515625e-05

Cross Validation Accuracy = 55.75%

C = 0.25 and Gamma = 0.000244140625

Cross Validation Accuracy = 55.75%

C = 0.25 and Gamma = 0.0009765625

Cross Validation Accuracy = 66.45%

C = 0.25 and Gamma = 0.00390625

Cross Validation Accuracy = 90.5%

C = 0.25 and Gamma = 0.015625

Cross Validation Accuracy = 91.45%

C = 0.25 and Gamma = 0.0625

Cross Validation Accuracy = 93.7%

C = 0.25 and Gamma = 0.25

Cross Validation Accuracy = 95.55%

C = 1 and Gamma = 6.103515625e-05

Cross Validation Accuracy = 55.75%

C = 1 and Gamma = 0.000244140625

Cross Validation Accuracy = 67.6%

C = 1 and Gamma = 0.0009765625

Cross Validation Accuracy = 90.6%

C = 1 and Gamma = 0.00390625

Cross Validation Accuracy = 91.3%

C = 1 and Gamma = 0.015625

Cross Validation Accuracy = 93.7%

C = 1 and Gamma = 0.0625

Cross Validation Accuracy = 95.7%

C = 1 and Gamma = 0.25

Cross Validation Accuracy = 97%

C = 4 and Gamma = 6.103515625e-05

Cross Validation Accuracy = 67.3%

C = 4 and Gamma = 0.000244140625

Cross Validation Accuracy = 90.65%

C = 4 and Gamma = 0.0009765625

Cross Validation Accuracy = 91.15%

C = 4 and Gamma = 0.00390625

Cross Validation Accuracy = 93.6%

C = 4 and Gamma = 0.015625

Cross Validation Accuracy = 94.75%

C = 4 and Gamma = 0.0625

Cross Validation Accuracy = 96.65%

C = 4 and Gamma = 0.25

Cross Validation Accuracy = 96.85%

C = 16 and Gamma = 6.103515625e-05

Cross Validation Accuracy = 90.65%

C = 16 and Gamma = 0.000244140625

Cross Validation Accuracy = 91.25%

C = 16 and Gamma = 0.0009765625

Cross Validation Accuracy = 93.75%

C = 16 and Gamma = 0.00390625

Cross Validation Accuracy = 94.85%

C = 16 and Gamma = 0.015625

Cross Validation Accuracy = 95.8%

C = 16 and Gamma = 0.0625

Cross Validation Accuracy = 96.75%

C = 16 and Gamma = 0.25

Cross Validation Accuracy = 96.25%

C = 64 and Gamma = 6.103515625e-05

Cross Validation Accuracy = 91.45%

C = 64 and Gamma = 0.000244140625

Cross Validation Accuracy = 93.4%

C = 64 and Gamma = 0.0009765625

Cross Validation Accuracy = 94.6%

C = 64 and Gamma = 0.00390625

Cross Validation Accuracy = 94.95%

C = 64 and Gamma = 0.015625

Cross Validation Accuracy = 96.7%

C = 64 and Gamma = 0.0625

Cross Validation Accuracy = 97.1%

C = 64 and Gamma = 0.25

Cross Validation Accuracy = 97.2%

C = 256 and Gamma = 6.103515625e-05

Cross Validation Accuracy = 93.55%

C = 256 and Gamma = 0.000244140625

Cross Validation Accuracy = 94.1%

C = 256 and Gamma = 0.0009765625

Cross Validation Accuracy = 94.3%

C = 256 and Gamma = 0.00390625

Cross Validation Accuracy = 95.35%

C = 256 and Gamma = 0.015625

Cross Validation Accuracy = 96.95%

C = 256 and Gamma = 0.0625

Cross Validation Accuracy = 96.5%

C = 256 and Gamma = 0.25

Cross Validation Accuracy = 97%

C = 1024 and Gamma = 6.103515625e-05

Cross Validation Accuracy = 94.2%

C = 1024 and Gamma = 0.000244140625

Cross Validation Accuracy = 95%

C = 1024 and Gamma = 0.0009765625

Cross Validation Accuracy = 94.75%

C = 1024 and Gamma = 0.00390625

Cross Validation Accuracy = 96.8%

C = 1024 and Gamma = 0.015625

Cross Validation Accuracy = 96.55%

C = 1024 and Gamma = 0.0625

Cross Validation Accuracy = 96.1%

C = 1024 and Gamma = 0.25

Cross Validation Accuracy = 96.65%

C = 4096 and Gamma = 6.103515625e-05

Cross Validation Accuracy = 94.7%

C = 4096 and Gamma = 0.000244140625

Cross Validation Accuracy = 94.3%

C = 4096 and Gamma = 0.0009765625

Cross Validation Accuracy = 95.15%

C = 4096 and Gamma = 0.00390625

Cross Validation Accuracy = 96.9%

C = 4096 and Gamma = 0.015625

Cross Validation Accuracy = 96.35%

C = 4096 and Gamma = 0.0625

Cross Validation Accuracy = 96.5%

C = 4096 and Gamma = 0.25

Cross Validation Accuracy = 96.6%

C = 16384 and Gamma = 6.103515625e-05

Cross Validation Accuracy = 94.3%

C = 16384 and Gamma = 0.000244140625

Cross Validation Accuracy = 94.7%

C = 16384 and Gamma = 0.0009765625

Cross Validation Accuracy = 95.8%

C = 16384 and Gamma = 0.00390625

Cross Validation Accuracy = 96.3%

C = 16384 and Gamma = 0.015625

Cross Validation Accuracy = 96.45%

C = 16384 and Gamma = 0.0625

Cross Validation Accuracy = 96.2%

C = 16384 and Gamma = 0.25

Cross Validation Accuracy = 96.9%

Average time = 0.188663634579

Best : C = 64, Gamma = 0.25, Accuracy = 97.2%

Best slected model : RBF kernel (based on cross validation for training)

Best : C = 64, Gamma = 0.25, Accuracy = 97.2%

For testing data:

Accuracy = 97.2% (1944/2000) (classification)

Collaborated with

I have collaborated with my project teammates Adarsha Desai and Mahesh Pottippala Subrahmanya