1.a) For a logistic function $h(w^T x) = \frac{1}{1+\exp -w^T x}$

We know that $P(Y = y_i | X = xj) = h(y_i w^T x_j)$

$$So\ if\ y_i = 1\ \ then\ h(y_i w^T x_j) = h(w^T xj)\ and\ if\ y_i = -1\ then\ h(y_i w^T x_j) = h(-w^T x_j)\ which\ \ can\ be\ rewritten\ as\ 1 - h(w^T xj)$$

Likelihood of $P(Y = y_i | X = xj)$, $\quad L(w) = \prod_{i=1}^{n} h(y_i w^T x_j)$

Log likelihood $l(w) = -\sum_{i=1}^{n} \log h(y_i w^T x_j)$

Error or Loss function = $-\sum_{i=1}^{n} \log h(y_i w^T x_j)$

1.b) Loss function = $-\sum_{i=1}^{n} \log h(y_i w^T x_j)$

= $-\sum_{i=1}^{n} \log \left(\frac{1}{1+\exp(-y_i w^T x_j)}\right)$

=$-\sum_{i=1}^{n} \log(1 + \exp(-y_i w^T x_j))$

Taking derivative of the above equation w.r.t w we get,

$\frac{1}{(1+\exp(-y_i w^T x_j))} * \exp(-y_i w^T x_i) * \ -y_i w^T x_j^T$

Which can be re written as $-\frac{y_i x_j^T}{\left(1+\exp\left(y_i w^T x_j^T\right)\right)}$

Taking second derivative we get $\left(\frac{y_i x_j^T}{\exp(-y_i w^T x_j)}\right)^2$ and we know square of anything is positive. Hence it is a convex function.

1.c)

When the training samples are linearly separable (i.e. the data can be perfectly classified by a linear classifier) the decision boundary must be parallel to the y-axis. Since we know that the logistic regression deals with the sigmoid function $\frac{1}{(1+\exp -w^T x)}$ so to make the sigmoid function a straight line w must be very very high. Hence we can say that when the data are linearly separable the value of w tens to go towards infinity.

1.d) Gradient $=\sum_n -\frac{y_i x_j^T}{\left(1+\exp\left(y_i w^T x_j^T\right)\right)} + 2\lambda w_k$

2.a) We need to show the non-convexity of $\left|\left|w\right|\right|_0$. So we know that value of L_0 norm is 0 for 0 and 1 for everything else.

Using the definition of Convex function which is if it satisfies,

$$f(ta + (1-t)b) \leq tf(a) + (1-t)f(b) \text{ for all t } [0,1] \text{ here, x1} = \text{a and x2} = \text{b}$$

then the function is said to be a convex function. Taking $L_0\ norm$ as our function $f(x)$ and a=1, b=0 and t=1/2 we get,

$$f\left(\frac{0+1}{2}\right) \leq \frac{f(0)+f(1)}{2}$$

$$1 \leq 0 + 1/2$$

which gives us $1 \leq 0.5$ which is a contradiction hence the function f(x) which is our L_0 norm is not a convex function.

2.b) we have to prove that the $\left|\left|w\right|\right|_1$ is convex in w.

We know that the L_1 norm of any number is the number itself. So for any two numbers a and b if we check them using the definition of convex function we get,

$f\left(\frac{a+b}{2}\right) \leq \frac{f(a)+f(b)}{2}$ and for $L_1 norm\ we\ know\ that\ \ f(a) = a\ and\ f(b) = b.$

So we get $\frac{a+b}{2} \leq \frac{a+b}{2}$ which satisfies the definition of the convex function. Hence we can see that L_1 norm is convex.

3.a) Writing the objective function in vector form, we get

$$(Y - Xw)^2 + \lambda\left|\left|w\right|\right|_2^2$$

$$= (y - xw)^T(y - xw) + \lambda w^T w$$

$$= (y^T - w^T x^T)(y - xw) + \lambda w^T w$$

$$= y^T y - y^T xw - w^T x^T y + w^T x^T xw + \lambda w^T w$$

Since, $w^T x^T y$ is a scalar we can take transpose of $w^T x^T y$ *and write is as* $y^T xw$

So this gives us $y^T y - 2y^T xw + w^T x^T xw + \lambda w^T w$

Taking derivative w.r.t w, we get $-2y^T x + 2w^T x^T x + 2\lambda w^T$

Equating this to zero we get $w = (xx^T + \lambda I_D)^{-1} x^T y$

3.b) When we apply a nonlinear feature mapping to all of these samples then the regularized least square function will be

$$J(w) = \frac{1}{2}\sum_n (y_n - w^T \phi(x_n)\text{^}2 + \frac{\lambda}{2}||w||_2^2$$

The derivative of this comes to $\sum_n (y_n - w^T \phi\,(x_n))(-\phi\,(x_n)) + \lambda w$

Equating it to zero we get $w = \sum_n \frac{1}{\lambda}(y_n - w^T \phi\,(x_n))(-\phi\,(x_n)) = \sum_n \alpha_n \phi(x_n) = \phi^T \alpha$----(1)

Where $\alpha = \frac{1}{\lambda}(y_n - w^T \phi\,(x_n))$.
Substituting the value of w in the J(w) function we get J as a function of α

$$J(w) = \frac{1}{2}||y - \phi w||_2^2 + \frac{\lambda}{2}||w||_2^2$$
$$= \frac{1}{2}||y - \phi\phi^T \alpha||_2^2 + \frac{\lambda}{2}||\phi^T \alpha||_2^2$$
$$= \frac{1}{2}||y - K\alpha||_2^2 + \frac{\lambda}{2}\alpha^T \phi\phi^T \alpha$$
$$= \frac{1}{2}\alpha^T K^T K\alpha - y^T K\alpha + \frac{\lambda}{2}\alpha^T K\alpha$$
$$= \frac{1}{2}\alpha^T K^2 \alpha - (Ky)^T \alpha + \frac{\lambda}{2}\alpha^T K\alpha$$
$$= J(\alpha)$$

$$J(\alpha) = \frac{1}{2}\alpha^T \phi\phi^T \phi\phi^T \alpha - (\phi\phi^T y)^T \alpha + \frac{\lambda}{2}\alpha^T \phi\phi^T \alpha$$

here, we can refer kernel matrix as $K = \phi\phi^T \epsilon R^{N*N}$ also we can do this because we have assumed K to be symmetric.

Taking the derivative of $J(\alpha)$ *we get* $\frac{\delta}{\delta\alpha}J(\alpha) = K^2\alpha - Ky + \lambda K\alpha$
Assuming that K is invertible we get $\alpha = (K + \lambda I)^{-1} y$

Substituting the value of $\alpha\ in\ opt\ w\ we\ get\ w^* = \phi^T(K + \lambda I)^{-1}y$.

3.c) Given a testing sample ϕ(x) we can predict it using our w^*

$$so\ \hat{y} = (w^*)^T\phi(x)$$

$$Substituting\ w^*\ we\ get\ \hat{y} = y^T(K + \lambda I)^{-1}\phi\phi(x)$$

3.d) whenever the feature space is very high we use kernel ridge regression as in that case it will transform the features into NxN kernel Matrix. So it performs better than linear ridge regression when feature space is high.

4.a) $Given\ k_3(x, x') = a_1k_1(x, x') + a_2k_2(x, x')$

For $K_3 to\ be\ a\ valid\ Kernel,\ x^TK_3x \geq 0\ for\ any\ vector\ x\ \epsilon R^N$

We get $x^T(a_1K_1 + a_2K_2)x = a_1x^TK_1x + a_2x^TK_2x$

Since $K_1 and\ K_2 are\ valid\ kernels\ x^TK_1x \geq 0\ and\ x^TK_2x \geq 0, the\ addition\ and\ non - negative\ combination\ of\ these\ two\ x^TK_3x\ is\ also\ \geq 0.$

Hence $K_3 is\ also$ a valid kernel.

4.b) $k_4(x, x') = f(x)\ f(x')$ where f (·) is a real valued function.
Assuming $f = \big(f(x_1), \dots\dots, f(x_1)\big)$ we can rewrite $K_4 = ff^T$, so $x^TK_4x = (x^Tf)^2$
Square of anything is positive. Hence K_4 is a valid kernel.

4.c) $k_5(x, x') = g\big(k_1(x, x')\big)$ where g is a polynomial function with positive coefficients.
A polynomial function can be represented as a product and summation of each element of the matrix K_1. Since $K_1 is\ a\ matrix\ with$ positive coefficient the product and summation of each element will also positive. That we have already proved in 4.a) and 4.d). Hence K_5 is a valid kernel.

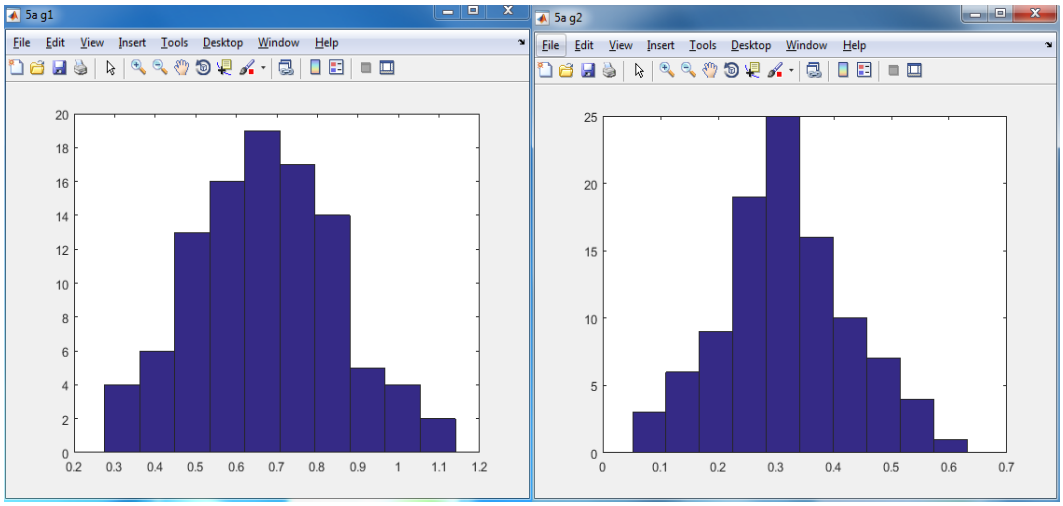
4.d) $k_6(x, x') = k_1(x, x')k_2(x, x')$

We know that the element by element product $K_1\ dot\ K_2$ gives the gram matrix. Suppose that K_1 is a covariance matrix of $(X_1, \dots\dots, X_n)$ and K_2 is the covariance matrix $(Y_1, \dots\dots, Y_n)$ then K becomes the covariance matrix of $(X_1Y_1, \dots\dots\dots, X_nY_n)$ which implies that K is symmetric and positive definite.

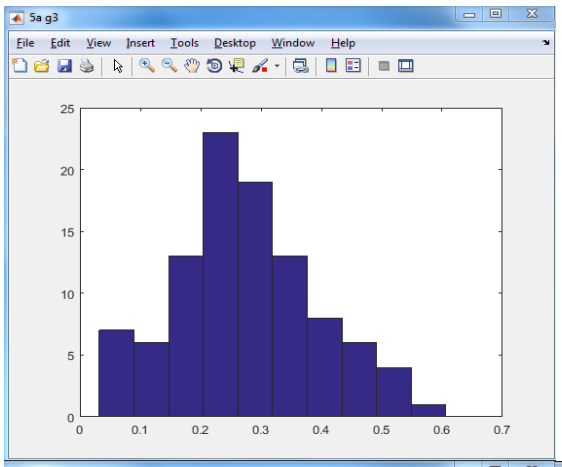
4.e) $K_7 = \exp\big(k_1(x, x')\big)$

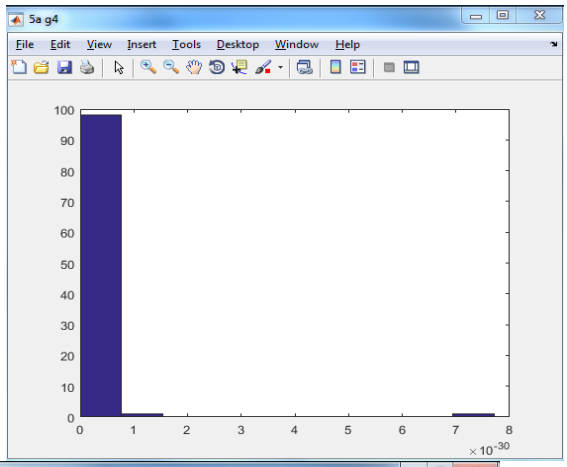
We can write $\exp(x) = \lim_{i\to\alpha}\left(1 + x + \frac{x^2}{2} + \cdots + \frac{x_i}{i_!}\right)$
Using the proff in 4.c) and the fact that $k(x, x') = \lim_{i\to\alpha} k_i(x, x')$ we can say that K_7 is a valid kernel.

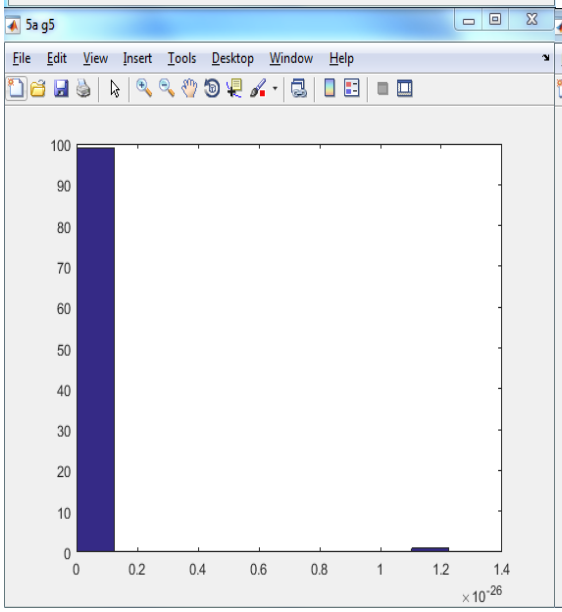
5a) Mean square error plots

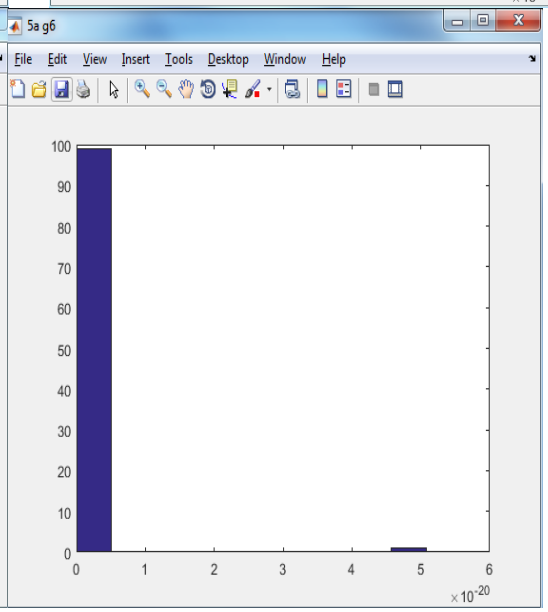


5.a) With 100 datasets each consiting of 10 samples.


5a g3
File Edit View Insert Tools Desktop Window Help
25
20
15
10
5
0
0 0.1 0.2 0.3 0.4 0.5 0.6 0.7

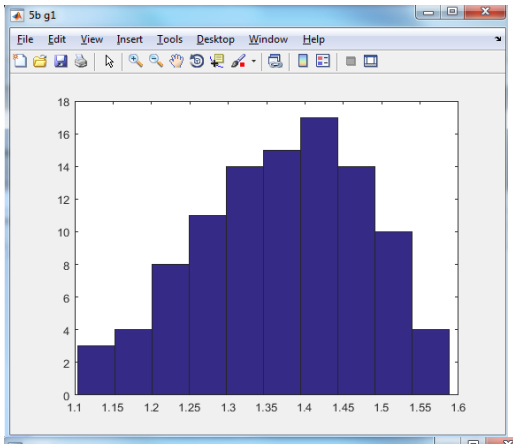

5a g4
File Edit View Insert Tools Desktop Window Help
100
90
80
70
60
50
40
30
20
10
0
0 1 2 3 4 5 6 7 8
×10-30

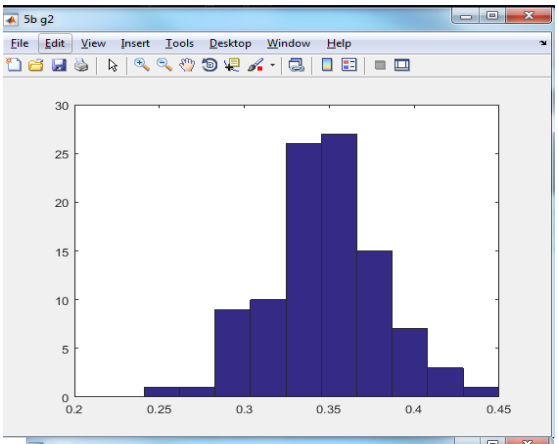

5a g5
File Edit View Insert Tools Desktop Window Help
100
90
80
70
60
50
40
30
20
10
0
0 0.2 0.4 0.6 0.8 1 1.2 1.4
×10-26

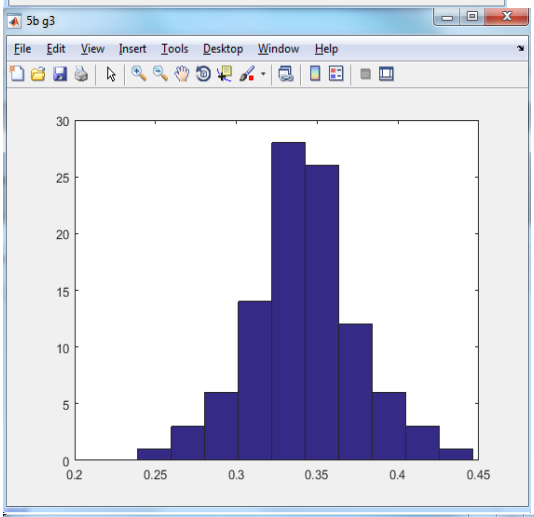

5a g6
File Edit View Insert Tools Desktop Window Help
100
90
80
70
60
50
40
30
20
10
0
0 1 2 3 4 5 6
×10-20

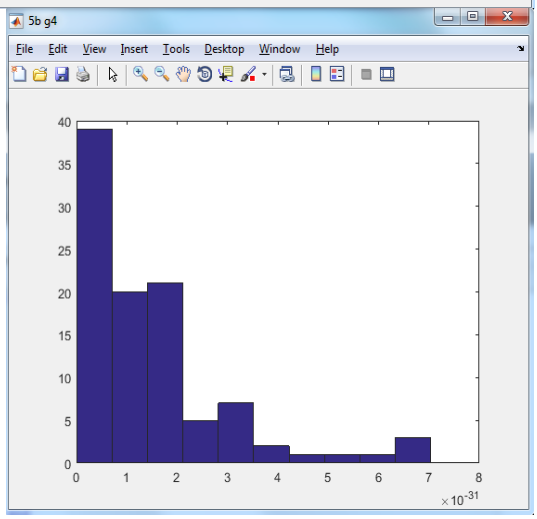
5b) 100 datasets with 100 samples Mean squared error plots

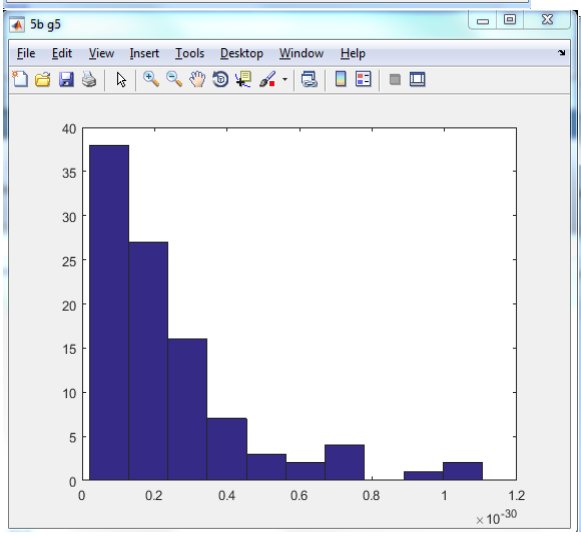
Susanth Dahal **USC id-7553849598**

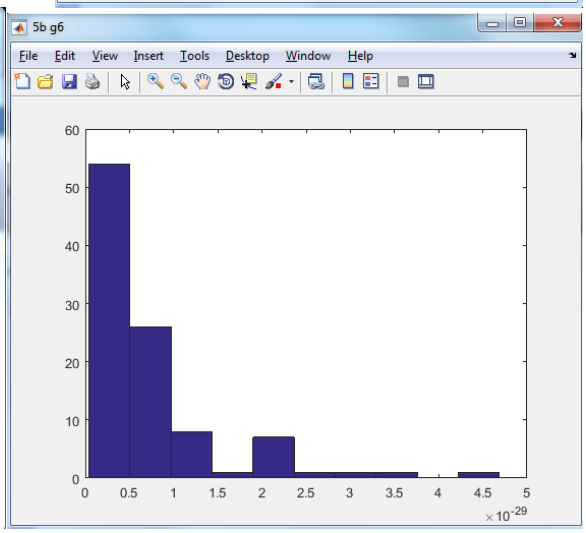












5.c)

For g1 bias=0.459804 and variance is 0.000000
For g2 bias=0.417701 and variance is 0.000000
For g3 bias=0.445293 and variance is 0.001623

For g4 bias=0.057818 and variance is 0.355233
For g5 bias=0.057818 and variance is 0.355233
For g6 bias=0.057818 and variance is 0.355233

Fifth b part

For g1 bias=0.467755 and variance is 0.000000
For g2 bias=0.798656 and variance is 0.000000
For g3 bias=0.805413 and variance is 0.000002
For g4 bias=0.447567 and variance is 0.350927
For g5 bias=0.447567 and variance is 0.350927
For g6 bias=0.447567 and variance is 0.350927

We can see that for each function g(x) as the model complexity increases the bias decreases and variance increases as it will try to fit each point. We can also infer that as the sample size increases the bias increases and variance decreases.

6.a) Now for the Linear kernel, Polynomial kernel and Gaussian Kernel I am using 3splits and 50/50 of the data.

Linear ridge

Split #1 opt Lambda=0.100000

Split #2 opt Lambda=0.010000

Split #3 opt Lambda=0.100000

The average error is 0.016470

Linear Kernel

Split #1 opt Lambda=0.100000

Split #2 opt Lambda=0.000000

Split #3 opt Lambda=0.010000

The average error is 0.016471

For Polynomial Kernel

Split #1 opt Lambda =0.00000 a=1.000000 c=3

Split #2 opt Lambda=0.010000 a=1.000000 c=3

Split #3 opt Lambda=0.010000 a=1.000000 c=3

The average error is 0.016285

For Gaussian Kernel

Split #1 opt Lambda=0.010000 sigma=8.000000

Split #2 opt Lambda=0.010000 sigma=8.000000

Split #3 opt Lambda=0.001000 sigma=8.000000

The average error is 0.016023

Yes linear ridge regression gives the same error as the kernel ridge regression with the linear kernel. Among the three kernels, Gaussian kernel performs the best.