Maxwell Jung
9319674

Homework 2

CMPSC 165B
October 30, 2022

1. $f(x|\sigma) = \frac{1}{2\sigma}e^{-\frac{|x|}{\sigma}}, -\infty < x < \infty, \sigma > 0$
   $p(X|\sigma) = \prod_{i=1}^{n} p(x_i|\sigma)$
   $\log(p(X|\sigma)) = \log(\prod_{i=1}^{n} p(x_i|\sigma))$
   $\log(p(X|\sigma)) = \sum_{i=1}^{n} \log(p(x_i|\sigma))$
   $\log(p(X|\sigma)) = \sum_{i=1}^{n} \log(\frac{1}{2\sigma}e^{-\frac{|x|}{\sigma}})$
   $\log(p(X|\sigma)) = \sum_{i=1}^{n}(\log(\frac{1}{2\sigma}) - \frac{|x|}{\sigma})$
   $f = \log(p(X|\sigma)) = -n\log(2\sigma) - \frac{\sum_{i=1}^{n}|x|}{\sigma}$
   $\frac{\partial f}{\partial \sigma} = \frac{-n}{\sigma} + \frac{\sum_{i=1}^{n}|x|}{\sigma^2} = 0$
   $-\sigma n + \sum_{i=1}^{n}|x| = 0$
   $\sigma = \frac{\sum_{i=1}^{n}|x|}{n}$

2. (a) $E_2(w) = ||Xw - y||^2 + \lambda||w||^2$
   $E_2(w) = (Xw - y)^T(Xw - y) + \lambda w^T w$
   $E_2(w) = w^T X^T Xw - y^T Xw - w^T X^T y + y^T y + \lambda w^T w$
   $E_2(w) = w^T X^T Xw - 2w^T X^T y + y^T y + \lambda w^T w$
   $\frac{\partial E_2(w)}{\partial w} = 2X^T Xw - 2X^T y + 2\lambda Iw = 0$
   $X^T Xw - X^T y + \lambda Iw = 0$
   $(X^T X + \lambda I)w = X^T y$
   $w^\star = (X^T X + \lambda I)^{-1} X^T y$

   (b) Since $X^T X$ is symmetric and λI is symmetric, $X^T X + \lambda I$ is also symmetric. If $X^T X + \lambda I$ can be proven to be positive definite, $X^T X + \lambda I$ is always invertible and $w^\star$ must exist.

   Proof that $X^T X + \lambda I$ is positive definitive:
   $y^T(X^T X + \lambda I)y = y^T X^T Xy + y^T \lambda Iy = y^T X^T Xy + y^T \lambda y$
   $= (Xy)^T Xy + \lambda y^T y = ||Xy||_2^2 + \lambda||y||_2^2$
   $||Xy||_2^2 + \lambda||y||_2^2 > 0$ since $\lambda > 0$

   The regularization term ensures that any singular matrix $X^T X$ becomes non-singular.

Homework 2

3. (a) To show that H is symmetric, we need to prove that $H = H^T$
$H^T = (X(X^TX)^{-1}X^T)^T$
$= X^{T^T}(X(X^TX)^{-1})^T$
$= X((X^TX)^{-1})^TX^T$
For any square matrix A, $(A^{-1})^T = (A^T)^{-1}$; therefore,
$= X((X^TX)^T)^{-1}X^T$
$= X(X^TX^{T^T})^{-1}X^T$
$= X(X^TX)^{-1}X^T = H$

(b) Proof by induction:
check $k = 1$ holds: $H^1 = H$
assume $k = n$ holds: $H^n = H^n = H$
show $k = n+1$ holds: $H^{n+1} = H^{n+1} = H$
$H^{n+1} = H^nH = HH = (X(X^TX)^{-1}X^T)(X(X^TX)^{-1}X^T)$
$= X(X^TX)^{-1}X^TX(X^TX)^{-1}X^T$
$= XI(X^TX)^{-1}X^T = X(X^TX)^{-1}X^T = H$

(c) Proof by induction:
check $k = 1$ holds: $(I-H)^1 = I-H$
assume $k = n$ holds: $(I-H)^n = I-H$
show $k = n+1$ holds: $(I-H)^{n+1} = I-H$
$(I-H)^{n+1} = (I-H)^n(I-H)$
$(I-H)^{n+1} = (I-H)(I-H)$
$= I^2 - HI - IH + H^2 = I - H - H + H = I - H$

(d) $trace(H) = trace(X(X^TX)^{-1}X^T)$
$= trace(X^TX(X^TX)^{-1}) = trace(I)$
I is the identity matrix of shape X^TX. Since the shape of X is N by $d+1$, shape of X^TX is $d+1$ by $d+1$. The trace of identity matrix is equal to it's column count which is $d+1$.

4. Python code. The weights of different regressions are compared using the l_2 norm of the difference.

   (a) least squares regression:
   GD vs SGD: 0.07789273851734434
   GD vs closed-form: 0.028082184297628626
   SGD vs closed-form: 0.07078212426409143

   (b) ridge regression:
   GD vs SGD: 0.0828492238148313
   GD vs closed-form: 0.028082139225422624
   SGD vs closed-form: 0.07580953531420018

Homework 2

5. Python code

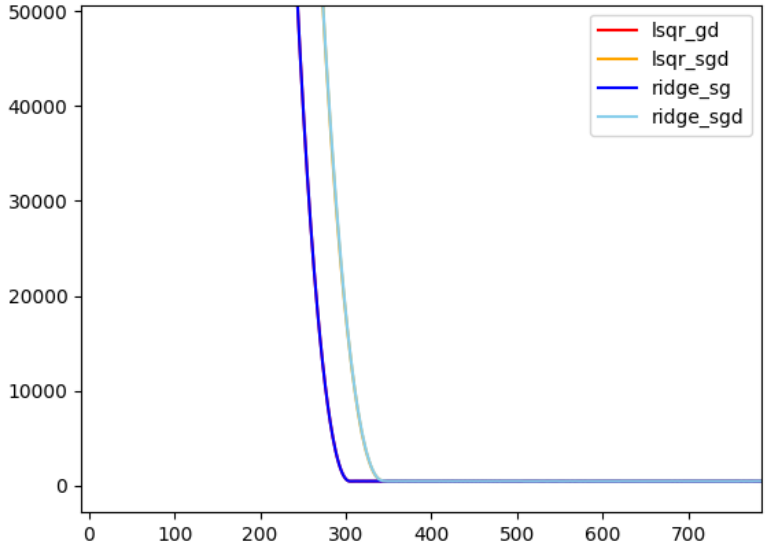

Figure 1: loss function over iteration count

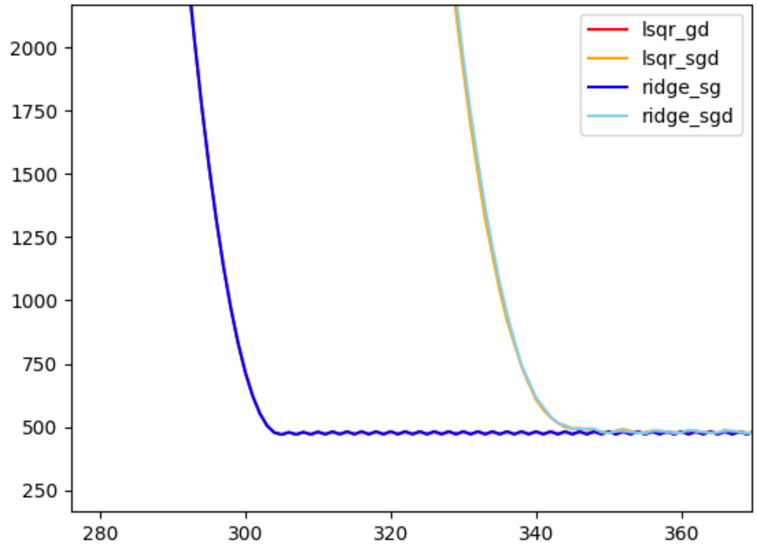

Figure 2: Figure 1 but zoomed in

6. Python code

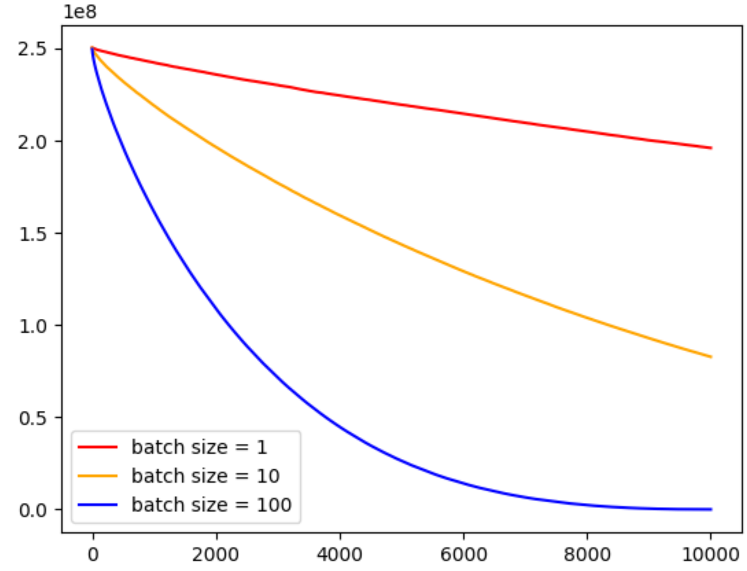

Figure 3: convergence for different batch size stochastic gradient descent

7. Python code

   (a) Effects of training data size on test error

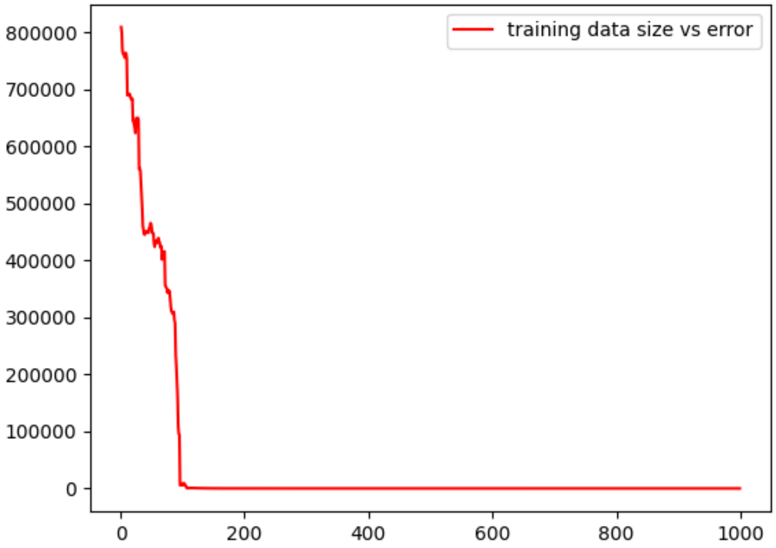

Figure 4: training data size vs error

Accuracy of regression is proportional to training data size. This intuitively makes sense because more data usually leads to better estimation.

(b) Effects of data dimension size on test error

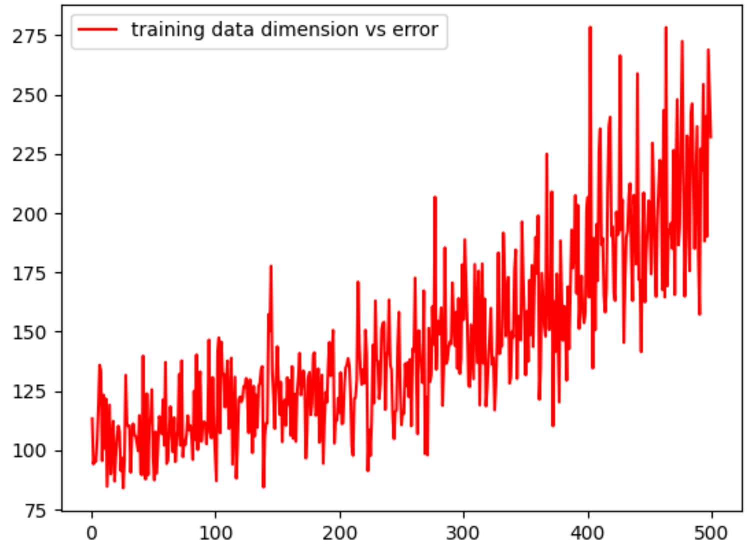

Figure 5: training data dimension vs error

Accuracy of regression is inversely proportional to training data dimension. This intuitively makes sense because the more features a data contains, the harder it is to form a generalized estimate.

8. Python code

   Effects of data dimension size on computation time (seconds)

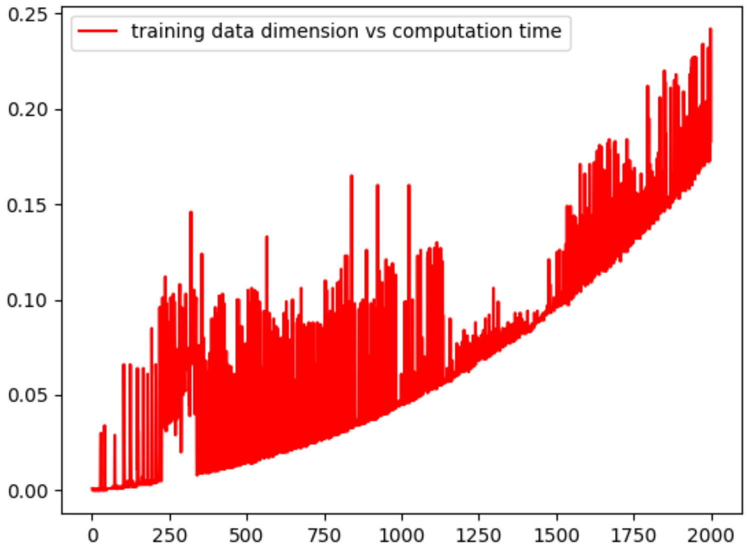

Figure 6: training data dimension vs error

Computation time increases with dimension size. This intuitively makes sense because the dimension size is directly proportional to the amount of data the algorithm needs to process.