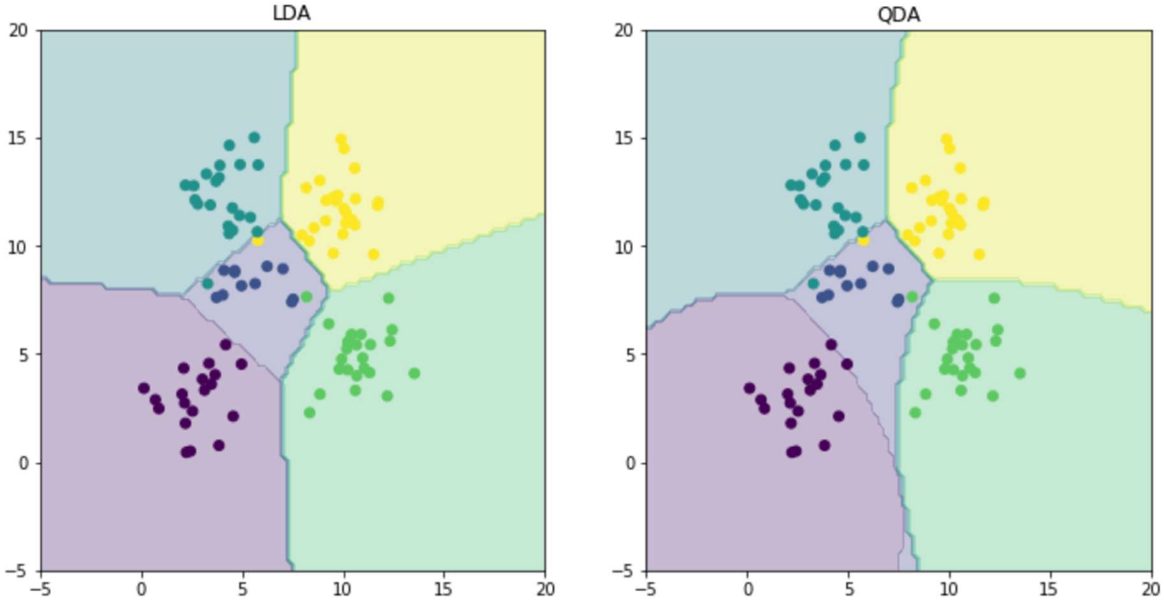
Classification and Regression

Anand anand6@buffalo.edu
Jing Chen jchen445@buffalo.edu

Linear v/s Quadratic discriminant analysis

```
LDA Accuracy = 0.97
QDA Accuracy = 0.96
```



The main difference of LDA and QDA is the covariance and the predicting of test data.

(1). In LDA, we create a 2 by 2 matrix of covariance corresponding to the 5 classes' covariance.

$$\Sigma = \frac{1}{N}\sum_{k=1}^{5} n_k \Sigma_k$$

The probability function is the following:

$$P(y_i) = \mu_i \Sigma^{-1} X_k^T - \frac{1}{2}\mu_i \Sigma^{-1} \mu_i^T + \log(P_i)$$

This produce 100 by 5 value of probability, then we choose the maximum probability value in each row and get the 100 by 1 matrix of corresponding class value.

(2) However, in QDA, we create a 5 by 2 by 2 covariances, so each class has its own covariance matrix.

we didn’t calculate a general covariance -> $\Sigma = \frac{1}{N}\sum_{k=1}^{5} n_k \Sigma_k$

Instead, we use -> Σ_k

The probability function is also different.

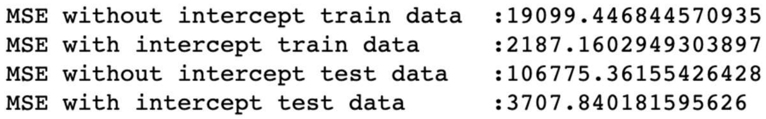
$$p(y_i) = -\frac{1}{2}\log\Sigma_i - \frac{1}{2}(x_k - \bar{x}_k)^T \Sigma_i^{-1} (x_k - \bar{x}_k) + \log(P_i)$$

So, the predictions of corresponding ytest are different.

Conclusion:

LDA assumes that each class has the common covariance because of which it results in the linear decision boundary. QDA assumes that each class its own covariance because of which the decision boundary encloses the class as closely as possible it becomes non-linear. So, it’s the reason for difference in LDA and QDA. LDA and QDA is used when we model regression as a classification and when the data is Gaussian.

OLS – Ordinary Least Squares - Linear Regression



```
MSE without intercept train data  :19099.446844570935
MSE with intercept train data     :2187.1602949303897
MSE without intercept test data   :106775.36155426428
MSE with intercept test data      :3707.840181595626
```

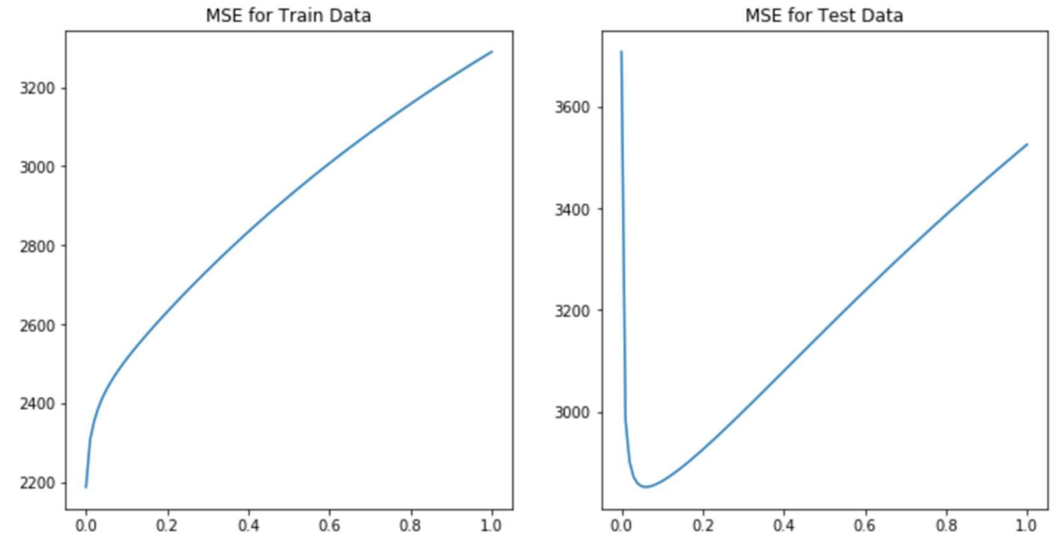
By not using an intercept, the error increased both on train data and test data. So linear fit starting from the origin which may be away from the density of data points and resulting in more deviation in the predicted response variable. Since the data do not have to always start from origin to fit the train data, having the intercept which can get the linear fit as closely to the density of data points as possible there by reducing the error in predicting the response variable.

Conclusion: Using intercept is better.

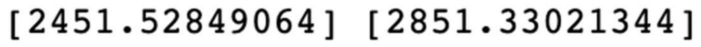
Ridge Regression

Ridge Regression is a shrinkage method and it shrinks the coefficients of the predictors based on the complexity parameter lambda. If lambda is zero then no regularization it considers all coefficients of predictors as it is but if lambda is increased it reduces the coefficient of the most insignificant predictors and improves the prediction of the response variable.

The plotted images are the following, as the lambda value grows, the MSE of train data grows up very quickly, while the MSE of test data decreases till optimal lambda value for the current data set and then increases.



Then we found the lambda value get the lowest error at 7th row with train and test MSE:



```
[2451.52849064] [2851.33021344]
```

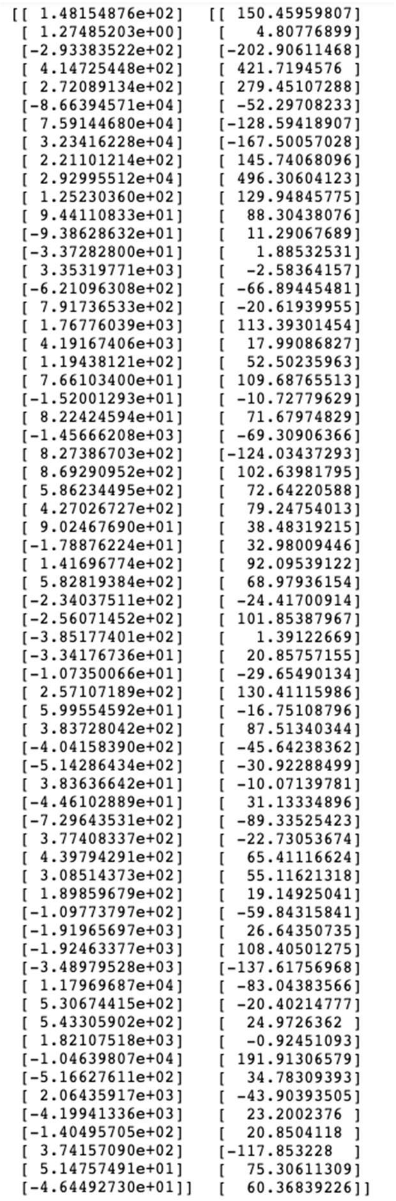
The value above are when lambda is **0.06** which is the optimal value of lambda to be used for the given dataset, we found although the train error is higher, the test error is lower than OLE.

Conclusion:

Since for ordinary linear regression, it assumes all variables are unbiased, there is no linear relation between themselves. It only finds a line fit the train data the best, so therefore, the variables may have high correlations and not fit test data very well. However, the Ridge regression, it assumes there may have correlation between variables and tries to reduce the correlation and regularize the coefficients as lambda steps in. Compared with the left array, the OLE weight, the ridge regression's weight, the right array have changed its weight, as the following shows, in 6th row, the weight changed from **-86640 to -52.**3, and in 7th row, the weight changed from **75914 to -128.6**.

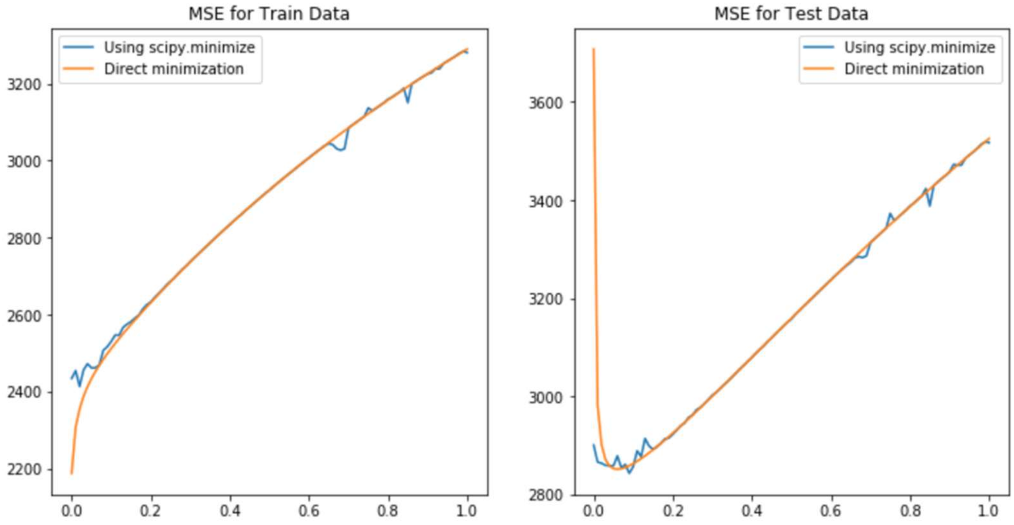
In conclusion, since the data set is variables of diabetes' features, it might have some correlation between themselves, and according to the test error performance, using ridge regression is better and the optimal lambda is 0.06.

The following two arrays are corresponding to the OLE and Ridge regression.



```
[[ 1.48154876e+02]  [[ 150.45959807]
 [ 1.27485203e+00]   [   4.80776899]
 [-2.93383522e+02]   [-202.90611468]
 [ 4.14725448e+02]   [ 421.7194576 ]
 [ 2.72089134e+02]   [ 279.45107288]
 [-8.66394571e+04]   [ -52.29708233]
 [ 7.59144680e+04]   [-128.59418907]
 [ 3.23416228e+04]   [-167.50057028]
 [ 2.21101214e+02]   [ 145.74068096]
 [ 2.92995512e+04]   [ 496.30604123]
 [ 1.25230360e+02]   [ 129.94845775]
 [ 9.44110833e+01]   [  88.30438076]
 [-9.38628632e+01]   [  11.29067689]
 [-3.37282800e+01]   [   1.88532531]
 [ 3.35319771e+03]   [  -2.58364157]
 [-6.21096308e+02]   [ -66.89445481]
 [ 7.91736533e+02]   [ -20.61939955]
 [ 1.76776039e+03]   [ 113.39301454]
 [ 4.19167406e+03]   [  17.99086827]
 [ 1.19438121e+02]   [  52.50235963]
 [ 7.66103400e+01]   [ 109.68765513]
 [-1.52001293e+01]   [ -10.72779629]
 [ 8.22424594e+01]   [  71.67974829]
 [-1.45666208e+03]   [ -69.30906366]
 [ 8.27386703e+02]   [-124.03437293]
 [ 8.69290952e+02]   [ 102.63981795]
 [ 5.86234495e+02]   [  72.64220588]
 [ 4.27026727e+02]   [  79.24754013]
 [ 9.02467690e+01]   [  38.48319215]
 [-1.78876224e+01]   [  32.98009446]
 [ 1.41696774e+02]   [  92.09539122]
 [ 5.82819384e+02]   [  68.97936154]
 [-2.34037511e+02]   [ -24.41700914]
 [-2.56071452e+02]   [ 101.85387967]
 [-3.85177401e+02]   [   1.39122669]
 [-3.34176736e+01]   [  20.85757155]
 [-1.07350066e+01]   [ -29.65490134]
 [ 2.57107189e+02]   [ 130.41115986]
 [ 5.99554592e+01]   [ -16.75108796]
 [ 3.83728042e+02]   [  87.51340344]
 [-4.04158390e+02]   [ -45.64238362]
 [-5.14286434e+02]   [ -30.92288499]
 [ 3.83636642e+01]   [ -10.07139781]
 [-4.46102889e+01]   [  31.13334896]
 [-7.29643531e+02]   [ -89.33525423]
 [ 3.77408337e+02]   [ -22.73053674]
 [ 4.39794291e+02]   [  65.41116624]
 [ 3.08514373e+02]   [  55.11621318]
 [ 1.89859679e+02]   [  19.14925041]
 [-1.09773797e+02]   [ -59.84315841]
 [-1.91965697e+03]   [  26.64350735]
 [-1.92463377e+03]   [ 108.40501275]
 [-3.48979528e+03]   [-137.61756968]
 [ 1.17969687e+04]   [ -83.04383566]
 [ 5.30674415e+02]   [ -20.40214777]
 [ 5.43305902e+02]   [  24.9726362 ]
 [ 1.82107518e+03]   [  -0.92451093]
 [-1.04639807e+04]   [ 191.91306579]
 [-5.16627611e+02]   [  34.78309393]
 [ 2.06435917e+03]   [ -43.90393505]
 [-4.19941336e+03]   [  23.2002376 ]
 [-1.40495705e+02]   [  20.8504118 ]
 [ 3.74157090e+02]   [-117.853228  ]
 [ 5.14757491e+01]   [  75.30611309]
 [-4.64492730e+01]]  [  60.36839226]]
```

Gradient descent for Ridge Regression

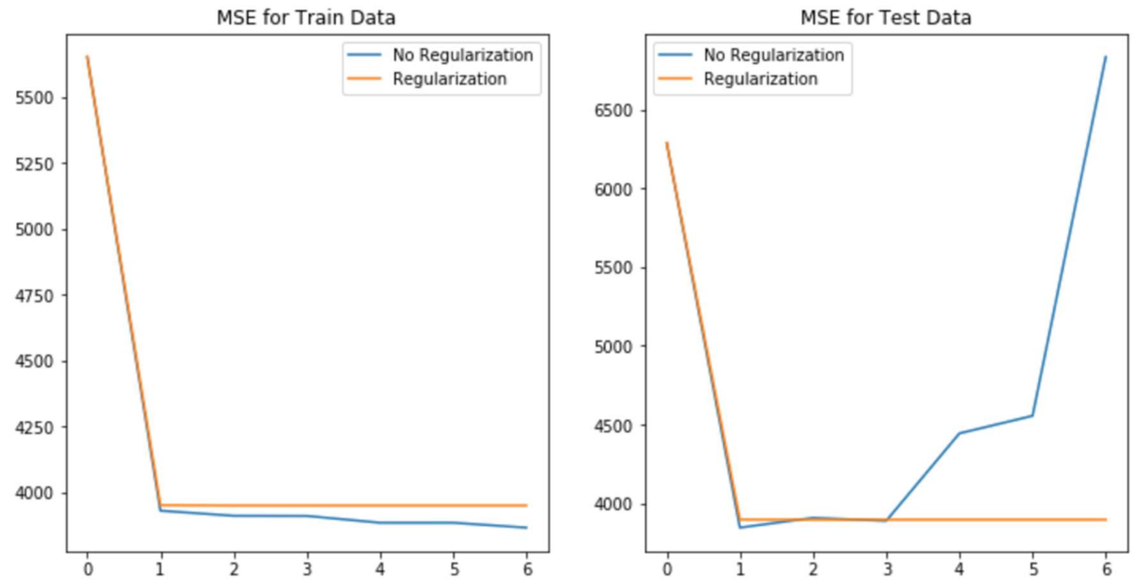


The blue line is representing for the gradient descent of ridge regression, the orange line is representing the ridge regression using direct minimization(using inverse of matrix). From the graphs, we can tell the movement of MSE for train data and test data are almost the same. However, when lambda is less than 0.3 and larger than 0.6, the MSE with gradient descent of ridge regression shows variations around the MSE with direct minimization of ridge regression.

	Direct Minimization		**Gradient Descent**	
Lambda	**MSE Train Data**	**MSE Test Data**	**MSE Train Data**	**MSE Test Data**
0.06	2451.528491	2851.330213	2469.9717	2847.316094
0.07	2468.077553	2852.349994	2473.999916	2841.787221

Optimal lambda found by gradient decent using test MSE is 0.07 as ridge regression is a purely a convex function and has only one global minima. The optimal lambda is between 0.06 and 0.07 and we will be able to find it with smaller step size but it will be slow convergence.

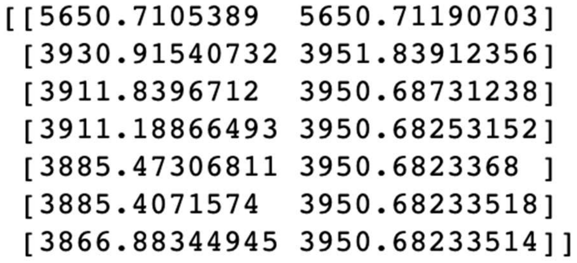
Higher degree polynomials v/s ridge regression(regularized)



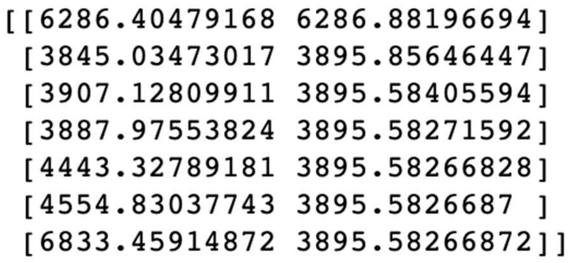
The blue line is where lambda equals 0 and orange line is where lambda equals 0.06.

As the graphs show, for train data, when p is 1, in both train data and test data, regularized and unregularized error reduced a lot and reach to very close points, but the unregularized error is lower than regularized error a little bit. As the p (the number of polynomials) gets larger, unregularized train error keeps lower than the train error of regularized. However, in the test data, the regularized error doesn't change much, while the unregularized error gets higher and when p larger than 3 its error grows exponentially. This is due to the regression overfitting, so even though the train error gets lower, the test error actually gets higher, as the selection of coefficients is based on train data.

train MSE: 0 vs 0.06 **test MSE: 0 vs 0.06**



```
[[5650.7105389  5650.71190703]
 [3930.91540732 3951.83912356]
 [3911.8396712  3950.68731238]
 [3911.18866493 3950.68253152]
 [3885.47306811 3950.6823368 ]
 [3885.4071574  3950.68233518]
 [3866.88344945 3950.68233514]]
```



```
[[6286.40479168 6286.88196694]
 [3845.03473017 3895.85646447]
 [3907.12809911 3895.58405594]
 [3887.97553824 3895.58271592]
 [4443.32789181 3895.58266828]
 [4554.83037743 3895.5826687 ]
 [6833.45914872 3895.58266872]]
```

According to the above results, for lambda is 0, the optimal p is 1, for lambda is 0.06 the optimal p is 4.

Comparison of different linear regression methods

For the given diabetes dataset, the regressions errors are shown below:

Regression types	Train MSE	Test MSE
OLE without intercept	19099.45	106775.36
OLE with intercept	2187.16	3707.84
Ridge regression lambda 0.06	2451.53	2851.33
Gradient Descent for Ridge lambda 0.07	2473.99	2841.78
Non-linear with lambda 0.06 and p=4	3950.68	3895.58

Comparing from the train error, the smallest MSE is OLE with intercept, while comparing from test error, the lowest one is gradient descent for ridge regression with lambda 0.07.If compare OLE with Ridge, even though the train error of Ridge regression is higher than OLE with intercept, the test error is much lower than OLE. If the train error doesn't have much difference, the test error should be a more important metric, as what we really care is regressions' ability in predicting the target value. The regressions MSE has revealed many things themselves, when choosing regression for predicting:

Conclusion:

- Choosing an intercept in OLE is important, it largely reduced both train and test error as OLE fits with a straight line, so whether limiting it to start from origin or not will change its performance huge.

- Choosing ridge regression can reduce the correlation between variables, if a user is not sure if the features in diabetes are correlated, he can use the ridge regression to see when lambda is not equals to 0 will the test error get lower.

- Choosing gradient descent in ridge regression is better than the original one, even though the MSE graph shows variations a little bit, its complexity is much lower than the ridge regression mathematic formula as when inverse of matrix is not guaranteed when the matrix is near to singular matrix. Not only because the inverse function will change the value a little bit, but also the runtime of ridge is longer, so if a user is using a very large dataset, the runtime of the gradient descent method is much shorter with proper step size.