LDA, QDA

1. what is the equation for linear and quadratic classifier found from year 1 data?

I’m not entirely sure how to interpret the coefficients, intercepts for LDA. I would have expected from the documentation two sets of coefficients and intercepts, on for each class. With only output from lda.intercept_ and lda.coef_, the natural interpretation is that the log odds is what can be determined from these outputs, so that a positive value indicates one class and a negative value indicates the other class. This interpretation holds up under numerical analysis.

GME: log(P(‘Red’)/P(‘Green’)) = [[-0.6935597 , -0.02651876]] * x + [0.43428967]

Using the raw covariance, means, and priors outputs the log posterior is very close to

```
-1/2*(np.subtract(x,means[0]) @ np.linalg.inv(cov[0]) @ \
                         (x-means[0])) + np.log(priors[0]) \
                      -1/2*np.log(np.linalg.det(cov[0]))
```

Where [0.52830189, 0.47169811] are priors for k = ‘Green’, ‘Red’ respectively,

```
[[ 3.10585179,  7.251883  ],
       [-1.986014  ,  4.76083117]]
```

are the means for k = ‘Green’, ‘Red’ respectively,

And

```
[[ 6.95308712,  2.91638275],
      [ 2.91638275, 14.1169589 ]]
```

is the covariance in the LDA case while

```
[array([[ 9.05321199,  6.30429963],
       [ 6.30429963, 23.2240902 ]]),
 array([[ 5.16987056, -0.65199185],
       [-0.65199185,  5.04784942]])]
```

are the covariances for ‘Green’, ’Red’ respectively in the QDA case.

Under numerical inspection with 1000 randomly generated values, this equation corresponds to the actual predictions at least 99.9% of the time.

For SPY, The priors, means, and covariances respectively are

```
(array([0.54716981, 0.45283019]),
 array([[ 0.64366897,  1.15382074],
       [-0.55140208,  1.89994944]]),
 [array([[0.37190062, 0.37929828],
        [0.37929828, 1.2288957 ]]),
  array([[ 0.58649164, -0.96430721],
        [-0.96430721,  3.18295322]])])
```

2. what is the accuracy for year 2 for each classifier. Which classifier is ”better”?
GME: LDA gives accuracy of .90, while qda gives accuracy of .81.   LDA is better.

SPY: LDA gives accuracy of .90, while qda gives accuracy of .81.   LDA is better.

3. compute the confusion matrix for year 2 for each classifier
GME: LDA: [[17, 1],[4, 30]], QDA: [[16, 2], [8, 26]]
SPY: (cm_linear, cm_quad) = (array([[31, 0], [5, 16]]), array([[31, 0], [10, 11]]))

4. what is true positive rate (sensitivity or recall) and true negative rate (specificity) for year 2?
GME: LDA: tpr, tnr = (0.88, 0.94) QDA: tpr, tnr = (0.76, 0.89)
SPY: tpr(cm_linear), tnr(cm_linear) = (0.76, 1.0)

5. implement trading strategies based on your labels for year 2 (for both linear and quadratic) and compare the performance with the ”buy-and-hold” strategy. Which strategy results in a larger amount at the end of the year?

GME: LDA: 49767.72 QDA: 8032.21, b&h:
SPY: LDA: 163.88 QDA: 158.74, b&h: 127.54

GaussianNB

1. implement a Gaussian naive bayesian classifier and compute its accuracy for year 2

GME: 0.77 SPY: .79

2. compute the confusion matrix for year 2

GME: [[14, 4], [8, 26]] SPY: [[31, 0], [11, 10]]

3. what is true positive rate and true negative rate for year 2

GME: tpr, tnr = (0.76, 0.78). SPY: tpr, tnr = (0.48, 1.0)

4. implement a trading strategy based on your labels for year 2 and compare the performance with the ”buy-and-hold” strategy. Which strategy results in a larger amount at the end of the year?

GME: 7091.39, 807.43, strategy better. SPY: 157.4, 127.54, strategy better

Student-t

1. implement a Gaussian naive bayesian classifier and compute its accuracy for year 2
2. compute the confusion matrix for year 2

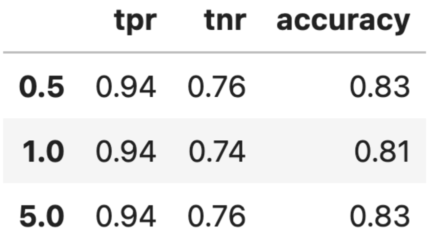
```
SPY: {0.5: array([[31,  0],
        [ 1, 20]]),
 1: array([[31,  0],
        [ 3, 18]]),
 5: array([[31,  0],
        [ 5, 16]])}

GME: {0.5: array([[17,  1],
        [ 8, 26]]),
 1: array([[17,  1],
        [ 9, 25]]),
 5: array([[17,  1],
        [ 8, 26]])}
```

[2

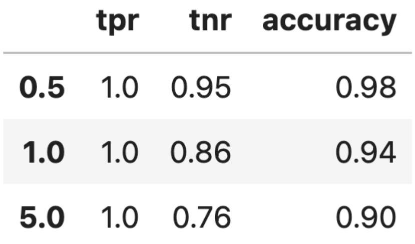
3. what is true positive rate and true negative rate for year 2

GME:



	tpr	tnr	accuracy
0.5	0.94	0.76	0.83
1.0	0.94	0.74	0.81
5.0	0.94	0.76	0.83

SPY:



	tpr	tnr	accuracy
0.5	1.0	0.95	0.98
1.0	1.0	0.86	0.94
5.0	1.0	0.76	0.90

4. implement a trading strategy based on your labels for year 2 and compare the performance with the "buy-and-hold" strategy. Which strategy results in a larger amount at the end of the year?

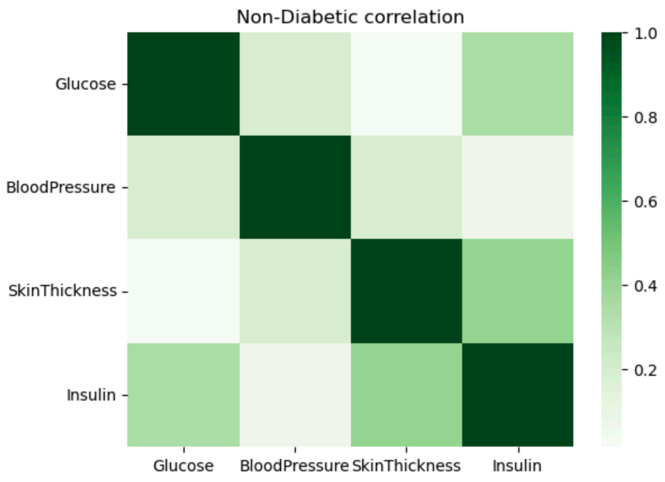
GME: 8891.48 vs 807.43. student-t NB is better.
SPY: 162.09 vs 127.54 student-t NB is better.

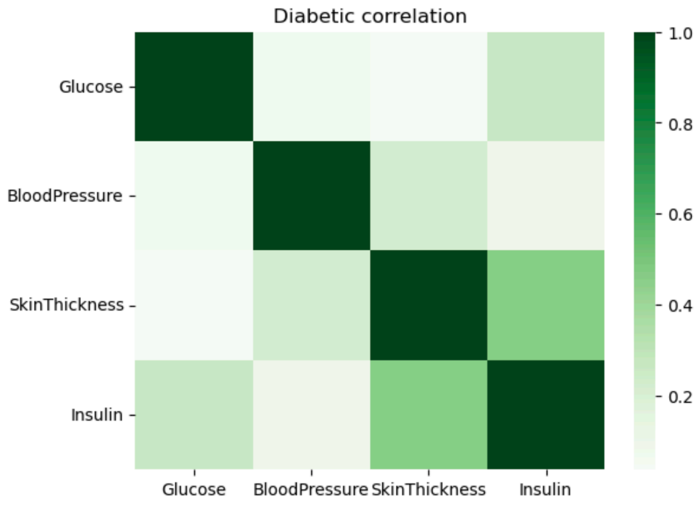
Diabetes

Question 1

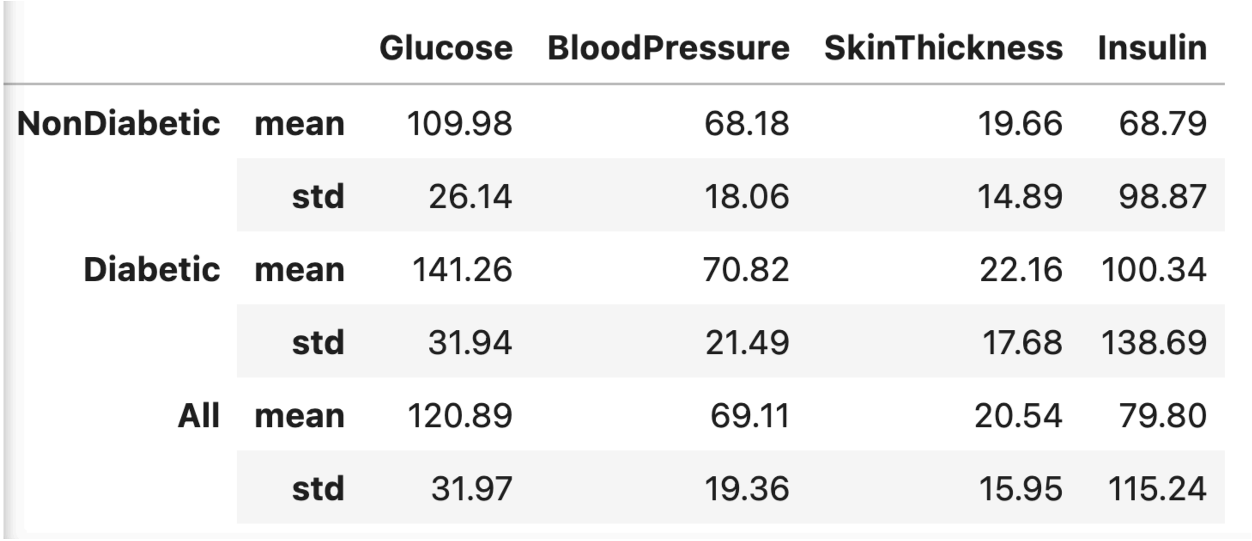
Plots:

Insulin & SkinThickness have the highest correlation for healthy as well as unhealthy people.





Skin thickness and glucose has the lowest correlation for both healthy and unhealthy people. Same results both cases.



		Glucose	BloodPressure	SkinThickness	Insulin
NonDiabetic	mean	109.98	68.18	19.66	68.79
	std	26.14	18.06	14.89	98.87
Diabetic	mean	141.26	70.82	22.16	100.34
	std	31.94	21.49	17.68	138.69
All	mean	120.89	69.11	20.54	79.80
	std	31.97	19.36	15.95	115.24

All metrics increase in mean and std when going from NonDiabetic to Diabetic

Question 3

1. apply Logistic regression

2. apply k-NN (k = 1, 3, 5)

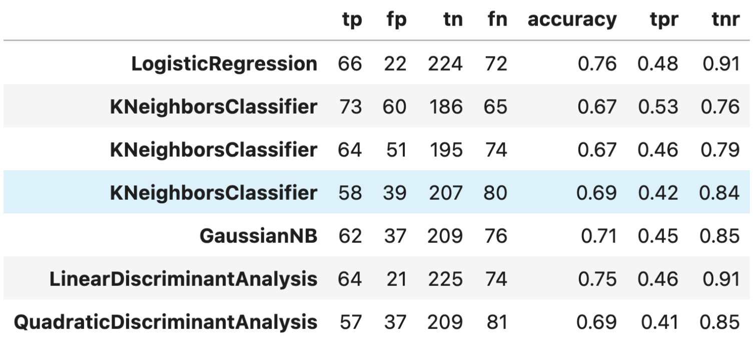
3. apply Naive-Bayesian classifier

4. apply Linear Discriminant

5. apply Quadratic Discriminant

6. compute its confusion matrices and summarize results

7. examine your results and correlation matrices. Any conclusions?



	tp	fp	tn	fn	accuracy	tpr	tnr
LogisticRegression	66	22	224	72	0.76	0.48	0.91
KNeighborsClassifier	73	60	186	65	0.67	0.53	0.76
KNeighborsClassifier	64	51	195	74	0.67	0.46	0.79
KNeighborsClassifier	58	39	207	80	0.69	0.42	0.84
GaussianNB	62	37	209	76	0.71	0.45	0.85
LinearDiscriminantAnalysis	64	21	225	74	0.75	0.46	0.91
QuadraticDiscriminantAnalysis	57	37	209	81	0.69	0.41	0.85

The LDA and Logistic Regression models result in the best accuracy and tnr, while KNeighbors is the worst for tnr but the best for tpr.