UNIVERSITY OF BIRMINGHAM

Machine Learning (Extended)

Computer Based Test 2 Bayesian Classification

November 9, 2016

Thomas Brereton 1708846

Table of Content

List of Figures

List of Tables

1 Abstract

In this computer based test we are asked to perform linear regression by least squares on a given data set. The dataset analysed is the Olympic men's running time (100m and 400m) for a given range of years.

In Task 1 we determine which polynomial function is the best fit for the 400m data. A common validation method is used to ensure the model is competent. In Task 2 we determine the best value of the regularisation factor, λ, for the polynomial functions of order 1 and 4.

In short, the results of Task 1 illustrate a polynomial function with order 2 best fits the Olympics men's 400m data. In Task 2, the results show a regularisation factor, λ, with value 0, gives the best predictive performance for both polynomial functions on the Olympic men's 100m data.

The analysis was performed in Matlab, and the code listings can be found in Appendices A and B.

2 Scope

The report consists of the following structure.

1. Overall Task

   Details of task set out in the "Computer Based Test 2."

2. Maximum Likelihood (ML)

   Discussion on training the ML with and without the Naive assumption

   Interpretation of results

3. Maximum A Posteriori (MAP)

   Discussion on training the MAP with and without the Naive assumption

   Interpretation of results

4. Maximum Likelihood VS Maximum A Posteriori

   Difference of ML and MAP

   Affects of Difference on this data

3 Overall Task

The analysis was performed on a data set of diseased and healthy patients. The data set has been plotted in Figure #.

4 Maximum Likelihood (ML)

4.1 Introduction

In Task 1, we are asked to analyse the Olympics men's 400m data. We must find the polynomial function of order **n** which best fits this data, where **n** is 1 to 4, and use 10-fold cross-validation to choose the "best" value of **n**. Refer to Appendix A for the Matlab code used for analysis.

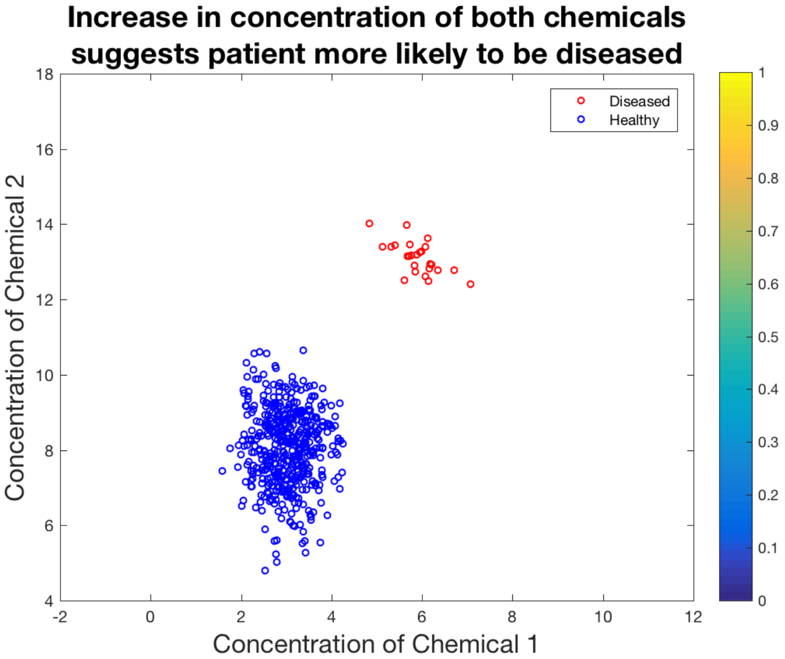


Figure 1: Comparison of the CV and Train loss for polynomial models of order **n**, where **n** = 0 to 4.

To compute the average cross validation loss, data (attributes and labels) are separated into 10 partitions (i.e. 10-fold), where 1 partition is reserved for testing the model. The 9 remaining partitions are used to learn the model with parameters, w_n. This models can use the attributes of the test partition to predict the labels, in this case the time ran for the men's 400m. The predicted values can then be compared to the actual values (labels) from the test partition. The cross validation (CV) loss is calculated by taking the mean squared difference (*msd*) of these two values. This is then repeated 9 more times by rotating the test partition through. The average of these 10 *msd* values is calculated and then plotted against the order to find the minimum value.

4.2 Results

From Figure 8 and Table 2 it is shown that a polynomial function with order 4 is the best fit. However, in order to determine the "best" order, it must be given a concrete meaning for this task. "Best" is considered to be the minimum mean squared loss, where loss is the difference between predicted and observed labels. Ideally, both cross-validation (CV) and training loss are considered but that is not always the case. So, three cases are defined as the following.

1. Only CV loss is considered
2. Only Train loss is considered
3. Both CV and Train loss are considered

Consider item 1, Figure 8 and Table 2 show that a polynomial function with order 2 is the best fit. It is visualised clearly on Figure 8 as the minimum point on the line. Looking at Table 2, a minimum value of 1.57 also corresponds with order 2.

Consider item 2, Figure 8 and Table 2 show that a polynomial function with order 4 is the best fit. Figure 8 shows a downward trend to the right, indicating that an order of 4 is indeed the best fit. Looking at Table 2, the minimum value, 3.30, also corresponds with an order of 4.

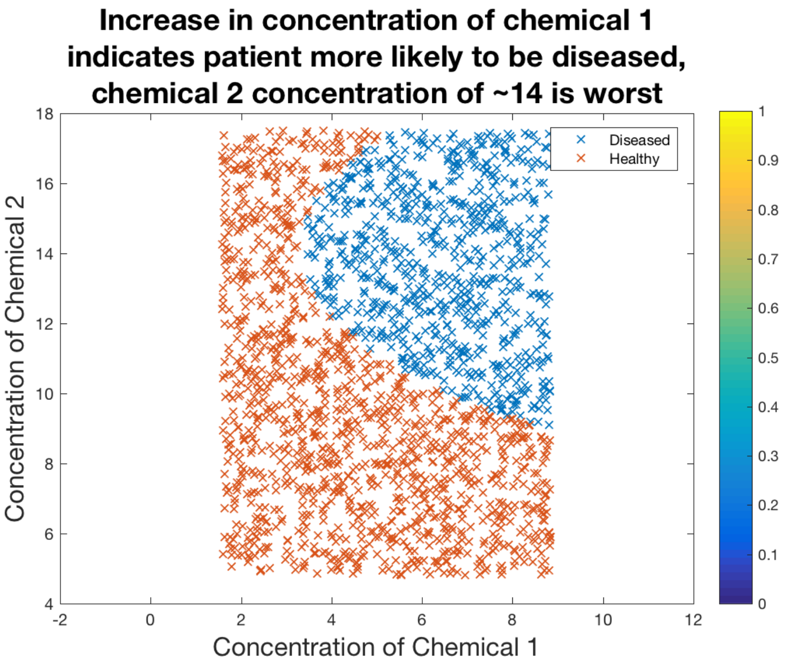


Figure 2: Comparison of the CV and Train loss for polynomial models of order **n**, where **n** = 0 to 4.

Table 1: Classification Differences between Methods

Order	CV Loss	Train Loss	Average Squared Loss
0	10.83	8.07	178.61
1	2.71	1.54	9.03
2	1.57	1.01	3.33
3	3.99	0.98	12.35
4	1.64	0.93	3.30

Consider item 3, Figure 8 and Table 2 show that a polynomial function with order 4 is the best fit. It is difficult to see this via the visualisation of Figure 8. However, looking at Table 2, the average squared loss (of CV and Train loss) is lowest when order equals 4.

Figure 11c shows how well each of the models fit the data. Interestingly, Figure **??**, shows an order of 4 provides an accurate model and more realistic future predictions than an order of 2. In comparison, an order of 2 shows that future predictions would increase in time at an increasing rate. This is highly unlikely given the downward trend of the data.

The remaining models in Figure 11 clearly do not model the data well, where orders 1 and 3 show that a time of 0 will be achieved soon. This is not humanly possibly so the models can be discarded.

4.3 Conclusion

The problem for Task 1 is to find the "best model based on average cross-validation loss." This only considers point 1 from before, therefore, a polynomial function with order 2 best fits the model based on average cross-validation loss.

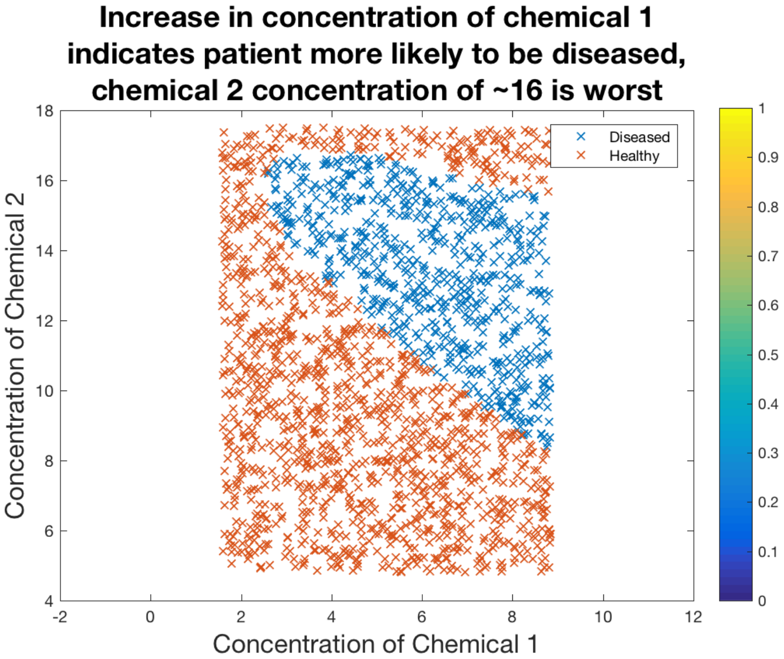
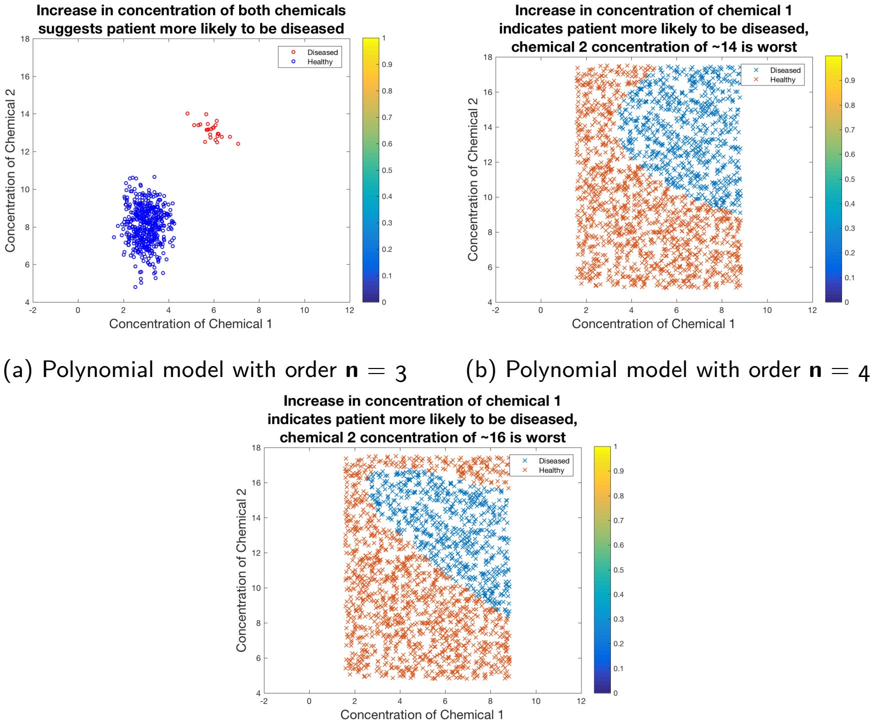


Figure 3: Comparison of the CV and Train loss for polynomial models of order **n**, where **n** = 0 to 4.



(a) Polynomial model with order **n** = 3

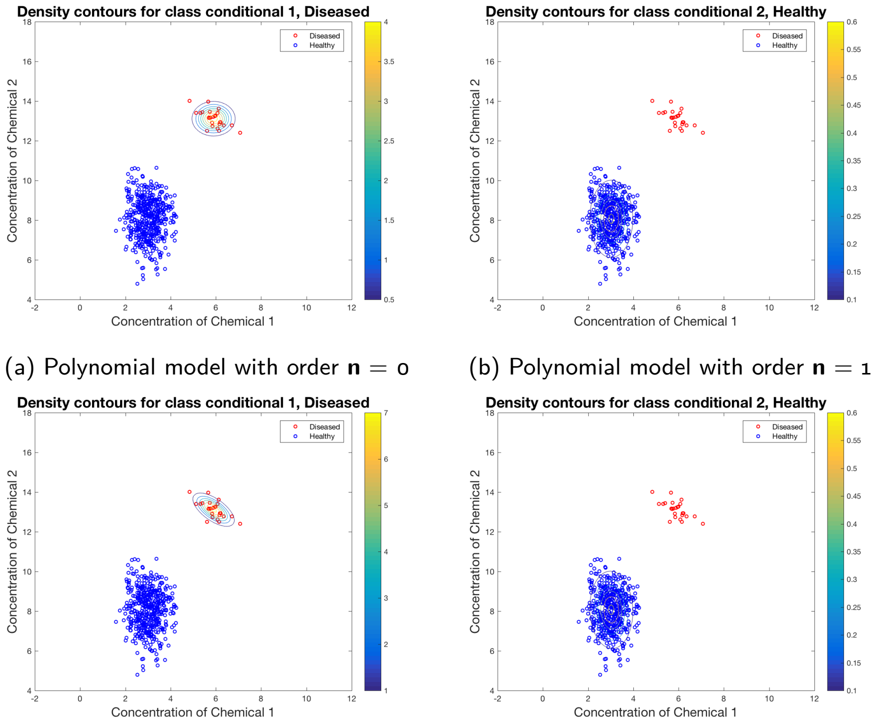
(b) Polynomial model with order **n** = 4

(c) Polynomial model with order **n** = 4

Figure 4: The original Olympic men's 400m data (blue crosses) with the polynomial models (red) overlaid without data standardisation.

4.4 Further Comments

Further investigation found if the data was not standardised, a systematic error is produced in the model for orders greater than or equal 4. Figure 9 illustrates an order of 3 produces no error, but an order of 4 does. Reviewing literature showed this systematic error is common when dealing with high order polynomial function[**WhenIsIt**].Therefore, it is good practice to standardise data when the regression model contains polynomial terms.



(a) Polynomial model with order **n** = 0

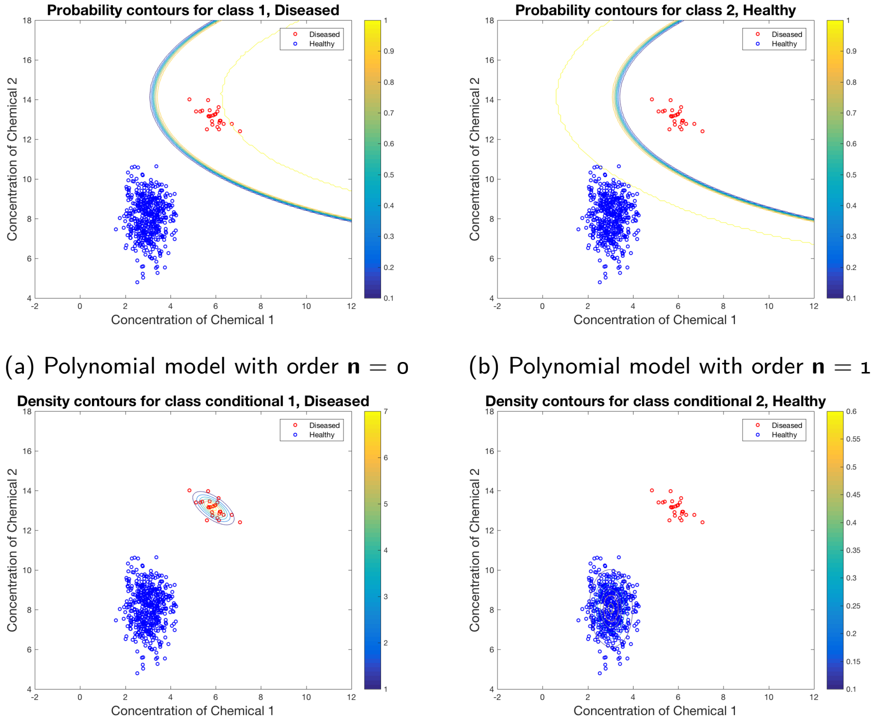
(b) Polynomial model with order **n** = 1

(c) Polynomial model with order **n** = 2

(d) Polynomial model with order **n** = 3

Figure 5: The original Olympic men's 400m data (blue crosses) with the polynomial models (red) overlaid.

Another point worth mentioning is the effect of permuting (randomising) the attributes. For this small dataset the permutations caused some instability in the model and different results were obtained as can be seen in Figure **??**. The results of the permuted data agree with the results from before.



(a) Polynomial model with order **n** = 0

(b) Polynomial model with order **n** = 1

(c) Polynomial model with order **n** = 2

(d) Polynomial model with order **n** = 3

Figure 6: The original Olympic men's 400m data (blue crosses) with the polynomial models (red) overlaid.

5 Maximum A Posteriori (MAP)

5.1 Introduction

In Task 1, we are asked to analyse the Olympics men's 400m data. We must find the polynomial function of order **n** which best fits this data, where **n** is 1 to 4, and use 10-fold cross-validation to choose the "best" value of **n**. Refer to Appendix A for the Matlab code used for analysis.

To compute the average cross validation loss, data (attributes and labels) are separated into 10 partitions (i.e. 10-fold), where 1 partition is reserved for testing the model. The 9 remaining partitions are used to learn the model with parameters, w_n. This models can use the attributes of the test partition to predict the labels, in this case the time ran for the men's 400m. The predicted values can then be compared to the actual values (labels) from the test partition. The cross validation (CV) loss is calculated by taking the mean squared difference (*msd*) of these two values. This is then repeated 9 more times by rotating the test partition through. The average of these 10 *msd* values is calculated and then plotted against the order to find the minimum value.

5.2 Results

From Figure 8 and Table 2 it is shown that a polynomial function with order 4 is the best fit. However, in order to determine the "best" order, it must be given a concrete meaning for this task. "Best" is considered to be the minimum mean squared loss, where loss is the difference between predicted and observed labels. Ideally, both cross-validation (CV) and training loss are considered but that is not always the case. So, three cases are defined as the following.

1. Only CV loss is considered
2. Only Train loss is considered
3. Both CV and Train loss are considered

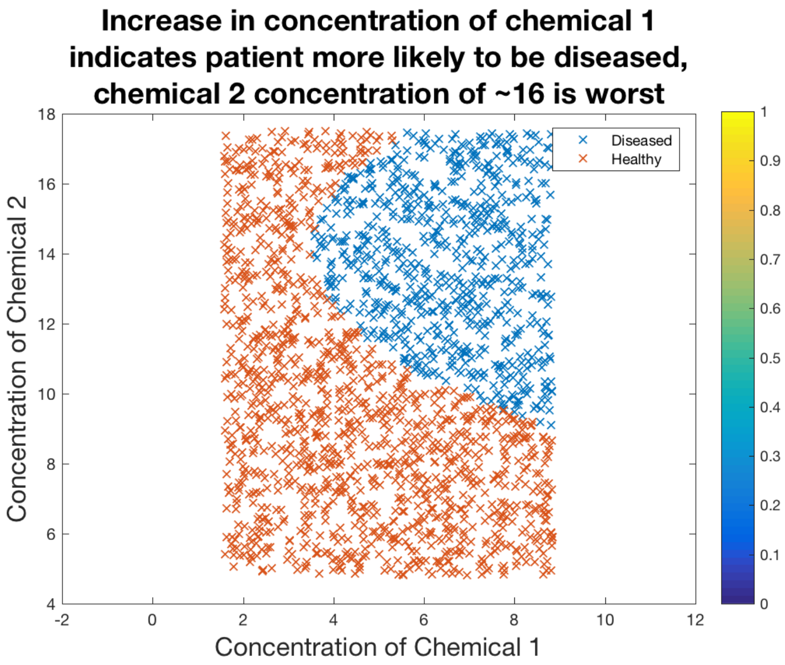


Figure 7: Comparison of the CV and Train loss for polynomial models of order **n**, where **n** = 0 to 4.

Table 2: Classification Differences between Methods

Order	CV Loss	Train Loss	Average Squared Loss
0	10.83	8.07	178.61
1	2.71	1.54	9.03
2	1.57	1.01	3.33
3	3.99	0.98	12.35
4	1.64	0.93	3.30

Consider item 1, Figure 8 and Table 2 show that a polynomial function with order 2 is the best fit. It is visualised clearly on Figure 8 as the minimum point on the line. Looking at Table 2, a minimum value of 1.57 also corresponds with order 2.

Consider item 2, Figure 8 and Table 2 show that a polynomial function with order 4 is the best fit. Figure 8 shows a downward trend to the right, indicating that an order of 4 is indeed the best fit. Looking at Table 2, the minimum value, 3.30, also corresponds with an order of 4.

Consider item 3, Figure 8 and Table 2 show that a polynomial function with order 4 is the best fit. It is difficult to see this via the visualisation of Figure 8. However, looking at Table 2, the average squared loss (of CV and Train loss) is lowest when order equals 4.

Figure 11c shows how well each of the models fit the data. Interestingly, Figure **??**, shows an order of 4 provides an accurate model and more realistic future predictions than an order of 2. In comparison, an order of 2 shows that future predictions would increase in time at an increasing rate. This is highly unlikely given the downward trend of the data.

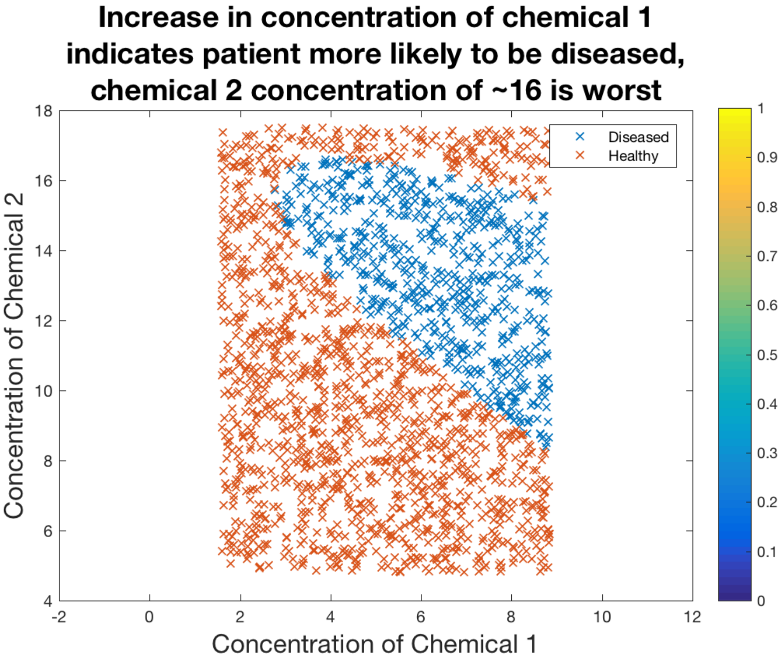


Figure 8: Comparison of the CV and Train loss for polynomial models of order **n**, where **n** = 0 to 4.

The remaining models in Figure 11 clearly do not model the data well, where orders 1 and 3 show that a time of 0 will be achieved soon. This is not humanly possibly so the models can be discarded.

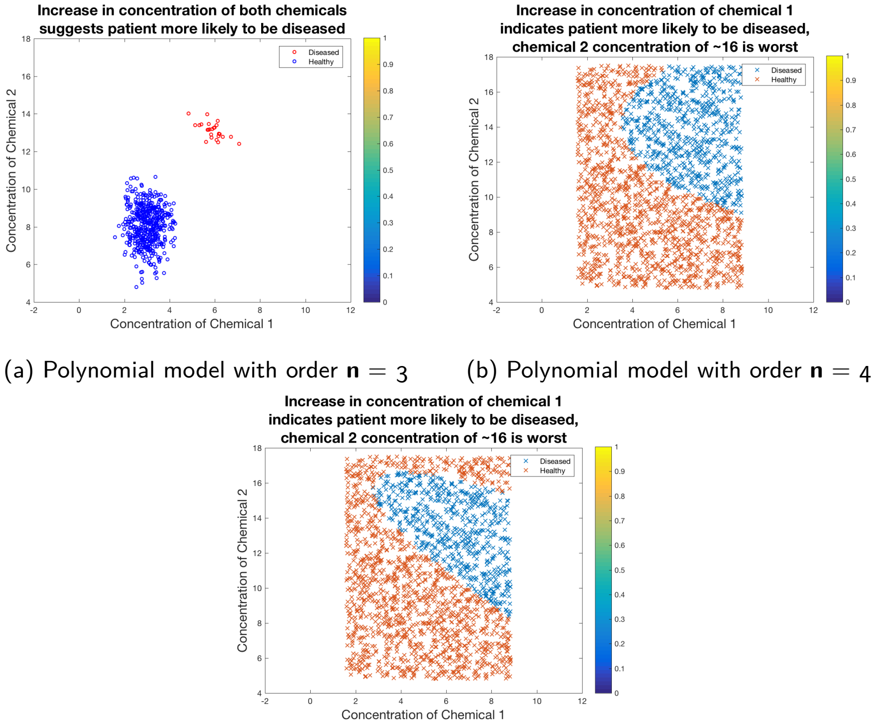
5.3 Conclusion

The problem for Task 1 is to find the "best model based on average cross-validation loss." This only considers point 1 from before, therefore, a polynomial function with order 2 best fits the model based on average cross-validation loss.

5.4 Further Comments

Further investigation found if the data was not standardised, a systematic error is produced in the model for orders greater than or equal 4. Figure 9 illustrates an order of 3 produces no error, but an order of 4 does. Reviewing literature showed this systematic error is common when dealing with high order polynomial function[**WhenIsIt**].Therefore, it is good practice to standardise data when the regression model contains polynomial terms.

Another point worth mentioning is the effect of permuting (randomising) the attributes. For this small dataset the permutations caused some instability in the model and different results were obtained as can be seen in Figure **??**. The results of the permuted data agree with the results from before.



(a) Polynomial model with order **n** = 3

(b) Polynomial model with order **n** = 4

(c) Polynomial model with order **n** = 4

Figure 9: The original Olympic men's 400m data (blue crosses) with the polynomial models (red) overlaid without data standardisation.

6 Maximum Likelihood vs Maximum A Posteriori (ML VS MAP)

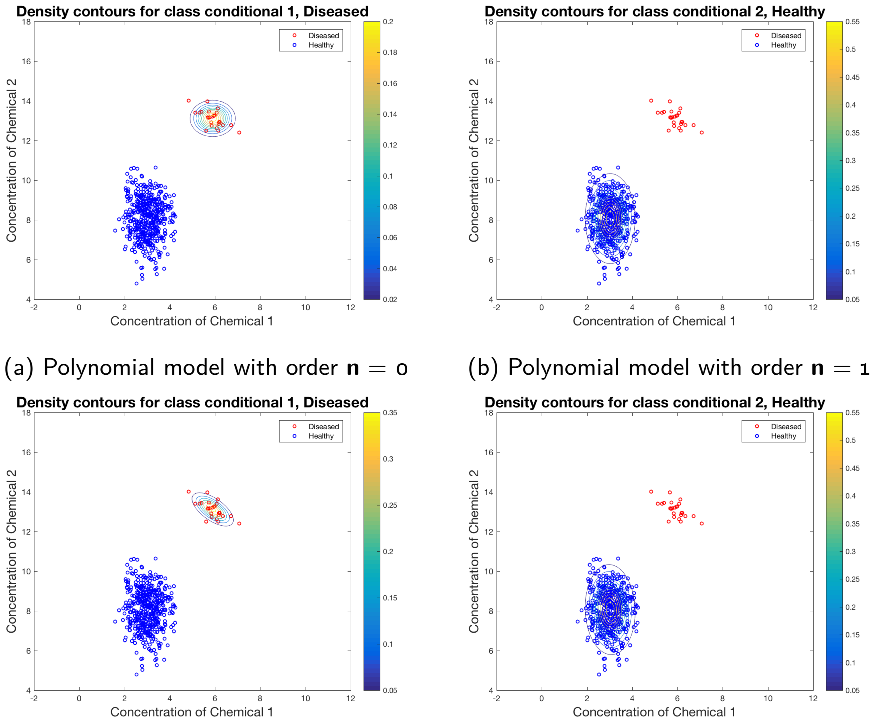
6.1 Difference of ML and MAP

In Task 1, we are asked to analyse the Olympics men's 400m data. We must find the polynomial function of order **n** which best fits this data, where **n** is 1 to 4, and use 10-fold cross-validation to choose the "best" value of **n**. Refer to Appendix A for the Matlab code used for analysis.

To compute the average cross validation loss, data (attributes and labels) are separated into 10 partitions (i.e. 10-fold), where 1 partition is reserved for testing the model. The 9 remaining partitions are used to learn the model with parameters, w_n. This models can use the attributes of the test partition to predict the labels, in this case the time ran for the men's 400m. The predicted values can then be compared to the actual values (labels) from the test partition. The cross validation (CV) loss is calculated by taking the mean squared difference (*msd*) of these two values. This is then repeated 9 more times by rotating the test partition through. The average of these 10 *msd* values is calculated and then plotted against the order to find the minimum value.

6.2 Affects of Difference on this Data

In Task 1, we are asked to analyse the Olympics men's 400m data. We must find the polynomial function of order **n** which best fits this data, where **n** is 1 to 4, and use 10-fold cross-validation to choose the "best" value of **n**. Refer to Appendix A for the Matlab code used for analysis.



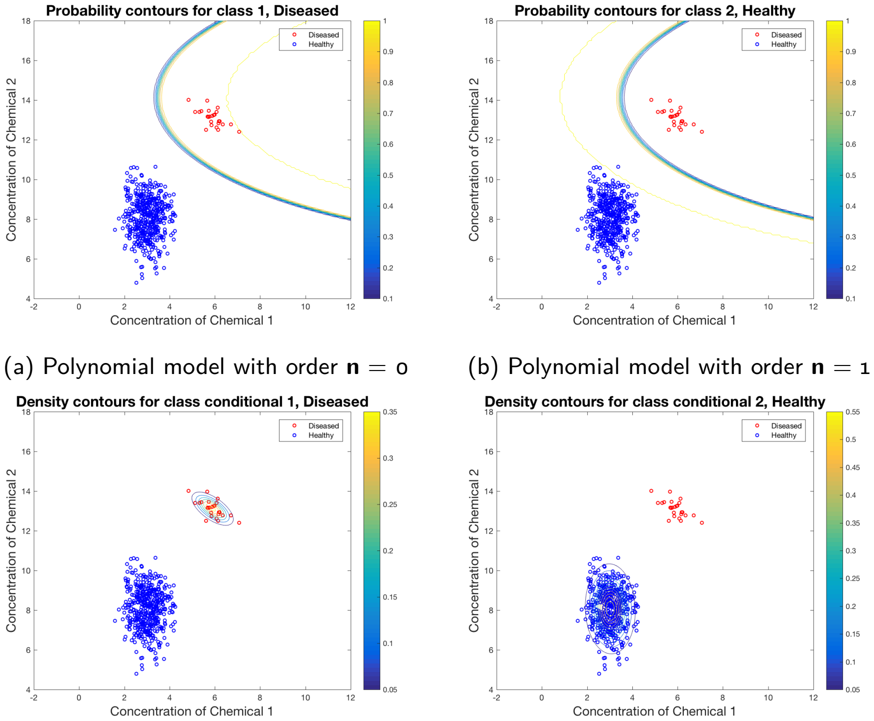
(a) Polynomial model with order **n** = 0 (b) Polynomial model with order **n** = 1

(c) Polynomial model with order **n** = 2 (d) Polynomial model with order **n** = 3

Figure 10: The original Olympic men's 400m data (blue crosses) with the polynomial models (red) overlaid.

To compute the average cross validation loss, data (attributes and labels) are separated into 10 partitions (i.e. 10-fold), where 1 partition is reserved for testing the model. The 9 remaining partitions are used to learn the model with parameters, w_n. This models can use the attributes of the test partition to predict the labels, in this case the time ran for the men's 400m. The predicted values can then be compared to the actual values (labels) from the test partition. The cross validation (CV) loss is calculated by taking the mean squared difference (*msd*) of these two values. This is then repeated 9 more times by rotating the test partition through. The average of these 10 *msd* values is calculated and then plotted against the order to find the minimum value.

Insert Table showing difference!!!



(a) Polynomial model with order **n** = 0

(b) Polynomial model with order **n** = 1

(c) Polynomial model with order **n** = 2

(d) Polynomial model with order **n** = 3

Figure 11: The original Olympic men's 400m data (blue crosses) with the polynomial models (red) overlaid.

7 References

Appendix A: Task 1

Appendix B: Task 2