IAML (Level 10) Assignment 1

Yifei Zhao

TOTAL POINTS

68.5 / 80

QUESTION 1

Question 1 : Linear Regression 22 pts

1.1 Data properties 3 / 3

✓ **- 0 pts Correct**

- 1 pts You failed to mention the correct range of the revision time attributes (2.72 to 48.01)

- 1 pts You failed to mention the correct range of the exam score targets (14.73 to 94.94)

- 1 pts You failed to mention the size of the dataset (50 datapoints)

- 1 pts You failed to mention that the attributes are continuous

- 3 pts You did not answer the question

- 1 pts Answer too long/answer box resized

- 0.5 pts Your reported values for the data ranges are incorrect

- 1 pts The range has been specified correctly, but the min and max values have been omitted.

1.2 Linear Model 2.5 / 3

- 0 pts Correct

- 2 pts Your reported model parameters are incorrect. They should be 17.898, and 1.441

- 1 pts You did not explain that the model parameters represented the intercept and slope

✓ **- 0.5 pts You included too many decimal places in your answer. Four or less is more appropriate**

- 3 pts You did not answer the question

- 1 pts Answer too long/answer box resized

- 0.5 pts Your explanation of the model parameters goes towards the right direction, but is not quite there: you could simply say the model parameters represent the intercept and slope

1.3 Display 3 / 3

✓ **- 0 pts Correct**

- 1 pts Your linear model is a poor fit to the data

- 1 pts The fitted line is not continuous

- 1 pts You did not label the axis

- 0.5 pts You did not add a legend

- 3 pts You did not answer the question

- 1 pts Answer too long/answer box resized

- 1 pts You should not represent your input data using a line

1.4 Custom implementation 3 / 3

✓ **- 0 pts Correct**

- 3 pts Your code is obviously wrong e.g. you did not perform the pseudo inversion

- 2 pts You did not implement the solution using basic matrix operations e.g. you used np.linalg.lstsq

- 1 pts Your is overly long and does not make use of numpy expressions e.g. not using np.dot or np.linalg.inv

- 3 pts You did not answer the question

- 1 pts Answer too long/answer box resized

- 2 pts You did not implement the solution using basic matrix operations e.g. you used np.linalg.solve

- 2 pts You implemented linear regression only for 1D input attributes.

- 3 pts You fitted a polynomial.

1.5 MSE 3 / 3

✓ **- 0 pts Correct**

- **2 pts** Missing square term
- **1 pts** y-hat and/or y should have a consistent lowerscript
- **2 pts** You added a square root term
- **1 pts** Missing a limitation e.g. distorted by outliers or not in the same units as the data
- **1 pts** You did not use the suggested notation for the ground truth and model prediction i.e. $$y$$ and $$\hat{y}$$
- **3 pts** You did not answer the question
- **1 pts** Answer too long/answer box resized
- **2 pts** Missing $$\frac{1}{N}$$ term

1.6 MSE 2 2 / 3

- **0 pts** Correct

✓ **- 0.5 pts** **You could comment on why MSEs are the same/different**

✓ **- 0.5 pts** **You included too many decimal places in your answer. Four or less is more appropriate**

- **1 pts** You MSE values is wrong. It should be 30.985
- **1 pts** For having different MSE scores for both approaches. They should be very similar for the most significant decimal places
- **3 pts** You did not answer the question
- **1 pts** Answer too long/answer box resized

1.7 Analysis 3 / 4

- **0 pts** Correct
- **2 pts** You failed to include the plot.
- **1 pts** The range of MSE is incorrect (should be around 8000 for $$w_1 = -2.0$$)
- **2 pts** The plot does not match the correct one in some aspect of the data
- **1 pts** The plot is unclear or insufficiently labelled i.e. you did not label the axis.
- **1 pts** You failed to identify the minimum value of MSE being close to 32.48 or the corresponding $$w_1$$ being close to 1.35

✓ **- 1 pts** **You failed to mention that the plot was convex/that there is a single minimum rather than multiple minima.**

- **0.5 pts** You have identified the plot as concave rather than convex.
- **1 pts** You failed to mention that the resulting minimum $$w_1$$ value was to be expected to be similar to one we found earlier (i.e. close to 1.4)/otherwise refer to the previously calculated result of $$w_1$$ in your argument.
- **0.5 pts** You included too many decimal places in your answer. Four or fewer is more appropriate.
- **1 pts** Answer too long/answer box resized
- **4 pts** You did not answer the question

QUESTION 2

Question 2 : Nonlinear Regression 18 pts

2.1 Polynomial regression 5 / 5

✓ **- 0 pts** **Correct**

- **2 pts** Your model predictions look very dfferent from the expected answer
- **1 pts** The plots for M = 3 and M = 4 are very different. In this example, they should look almost identical.
- **2 pts** You did not plot a continuous line for the models
- **1 pts** You did not plot the input data
- **1 pts** Your plot is unclear e.g. missing/wrong axes or a legend
- **5 pts** You did not answer the question
- **1 pts** Answer too long/answer box resized
- **0.5 pts** The input data should be plotted as points as there is no connection among them
- **3 pts** You did not plot some of the lines for the models

2.2 Bar plot 3 / 3

✓ **- 0 pts** **Correct**

- **2 pts** Your bar plot looks very different from the expected answer e.g. M = 3 and M = 4 should have very similar values for MSE, but both are less than M = 1

or M = 2

- **1 pts** Your estimated MSE values are incorrect e.g. M = 1 should be around 24.7
- **1 pts** Your plot is not clear e.g. missing axes
- **3 pts** You did not answer the question
- **1 pts** Answer too long/answer box resized

2.3 Analysis 3 / 4

- **0 pts** Correct
- **1 pts** You failed to mention that M = 3 and M = 4 give very similar predictions.
- **2 pts** You said that the M = 4 model is better. The lower parameter M = 3 is a good trade off between complexity and performance i.e. it requires less parameters.
- **1 pts** You did not mention that the weight vector entry corresponding to x 4 i is very small.
- **4 pts** You did not answer the question
- **1 pts** Answer too long/answer box resized

✓ **- 1 pts you did not explain correctly why M = 3 model is better. The lower parameter M = 3 is a good trade off between complexity and performance i.e. it requires less parameters.**

- **2 pts** you did not answer which model to choose. The lower parameter M = 3 is a good trade off between complexity and performance i.e. it requires less parameters.
- **1 pts** you did not explain correctly why M=3 is better. You don't have sufficient evidence to use overfitting as an argument. To comment on overfitting a model should be evaluated on heldout/unseen data. The lower parameter M = 3 is a good trade off between complexity and performance i.e. it requires less parameters.

2.4 RBF 5 / 6

- **0 pts** Correct
- **3 pts** Your plot looks very different from the expected answer
- **2 pts** You only plotted your model predictions where the input data was. You should have used more input points to better visualize the predictions i.e. to make the plot more continuous
- **1 pts** You failed to mention what would happen if the width parameter is too large i.e. more datapoints further away from the basis center are included resulting in underfitttting (or overly smooth predictions)

✓ **- 1 pts You failed to mention what would happen if the width parameter is too small i.e. model predictions will be a constant for points that are not close to the kernel center**

- **1 pts** Your plot is not very clear e.g. you did not plot the input data, it is missing a legend, axis labels or the plot is not complete
- **6 pts** You did not answer the question
- **1 pts** Answer too long/answer box resized
- **3 pts** You failed to provide the figure
- **2 pts** Some of the plotted graphs are different from the expected answer

QUESTION 3

Question 3 : Decision Trees 26 pts

3.1 Dataset analysis 4 / 4

✓ **- 0 pts Correct**

- **1 pts** You reported the wrong number of attributes. It should have been 136
- **0.5 pts** You did not report the correct train set size (4800)
- **0.5 pts** You did not report the correct test set size (1200)
- **0.5 pts** You did not report the correct number of positive labels in the train set (2335)
- **0.5 pts** You did not report the correct number of negative labels in the train set (2465)
- **0.5 pts** You did not report the correct number of positive labels in the test set (592)
- **0.5 pts** You did not report the correct number of

negative labels in the test set (608)

- **4 pts** You did not answer the question
- **1 pts** Answer too long/answer box resized
- **1 pts** You only reported the ratio of smiling faces to total faces, not the total number of smiling and not smiling faces.
- **1 pts** You only reported the ratio of smiling faces to not smiling faces, not the total number of smiling and not smiling faces.

3.2 Analysis 4 / 4

✓ **- 0 pts Correct**

- **2 pts** You failed to mention any difference between the two sets of points e.g. the mouth is wider and the chin is lower on the smiling face
- **1 pts** You did not mention the difference clearly
- **3 pts** The plot does not contain two faces
- **1 pts** You did not plot both faces on the same figure. This makes it difficult to see the differences
- **1 pts** You did not include a legend or way to indicate which points correspond to the train set and which correspond to the test
- **1 pts** You plot is missing axes labels, you plotted lines instead of points, etc.
- **4 pts** You did not answer the question
- **1 pts** Answer too long/answer box resized

3.3 Decision Trees 1 / 2

- **0 pts** Correct
- **1 pts** You did not specify the correct measure used by sklearn by default i.e. gini

✓ **- 1 pts You did not specify an advantage of gini over entropy e.g. computing entropy requires more computation as you need to take logarithms**

- **2 pts** You did not answer the question
- **1 pts** Answer too long/answer box resized

Incorrect, gini and entropy are equivalent in performance (and misclassification). But one advantage is that gini is more computationally efficient as it does not need log computations.

3.4 DT Depth 3 / 3

✓ **- 0 pts Correct**

- **1 pts** You did not mention what happens when you use a maxiumum depth that is too small e.g. underfitting
- **2 pts** You did not give two examples of what happens when you use a maxiumum depth that is too large e.g. overfitting, large trees requiring larger storage, and can be slow to evaluate
- **1 pts** You only gave one example of what happens when you use a maxiumum depth that is too large e.g. overfitting, large trees requiring larger storage, and can be slow to evaluate
- **3 pts** You did not answer the question
- **1 pts** Answer too long/answer box resized

3.5 Hyperparameter tuning 6 / 6

✓ **- 0 pts Correct**

- **2 pts** Your reported accuracy numbers are significantly different from what is expected. Are you sure you used the correct random seed and the correct version of sklearn?
- **1 pts** You did not report the train set accuracy
- **1 pts** You did not report the test set accuracy
- **2 pts** You did not correctly identify why max_depth = 8 is the better model i.e. you did not mention the overfitting that happens in the case of max_depth = 20 which clarifies why max_depth = 8 is best.
- **1 pts** You reported results with too many digits after the decimal place. Less than three would have been sufficient.
- **6 pts** You did not answer the question
- **1 pts** Answer too long/answer box resized

3.6 Attribute importance 3 / 5

- **0 pts** Correct
- **2 pts** Your reported attributes are different from

the expected answer of \`x50' \`y48', and \`y29'

- **1 pts** The order of your attributes is incorrect. It should be \`x50' \`y48', and \`y29'.

- **2 pts** You reported the indices of the attributes and not their names i.e. 100, 97, and 59 instead of \`x50' \`y48', and \`y29'.

✓ **- 2 pts** **You did not give a reason why the attributes were likely to be good choice in the context of the task. The most important attribute, \`x50', corresponds to the upper right lip. This is a part of the face that is likely to move a lot when someone smiles.**

- **5 pts** You did not answer the question

- **1 pts** Answer too long/answer box resized

- **1 pts** You answered that the most important attribute does not make sense. \`x50' corresponds to the upper right lip and it is a part of the face that is likely to move a lot when someone smiles.

- **1 pts** You reported one incorrect attribute. The expected answer was \`x50' \`y48', and \`y29'.

3.7 Analysis 2 / 2

✓ **- 0 pts** **Correct**

- **2 pts** You did not give a sensible limitation of the choice of input feature encod-
ings e.g. they are based on absolute pixel locations, it might be better to
use relative distances between points or they are subject to noise if detected
poorly.

- **1 pts** In your answer, you incorrectly said that the data only contains frontal views
of faces, it actually contains side views as well

- **2 pts** You did not answer the question

- **1 pts** Answer too long/answer box resized

QUESTION 4

Question 4 : Evaluating Binary Classifiers 14 pts

4.1 Classification accuracy 2 / 4

- **0 pts** Correct

- **2 pts** Your accuracy numbers are incorrect (or missing)

- **1 pts** You failed to state that alg_1 has the best performance

- **1 pts** You reported numbers in the range of 0 to 1. You should have used 0 to 100
as the questions asked for %

✓ **- 1 pts** **Your answer is too generic or you failed to mention a limitation of using a FIXED threshold i.e. the threshold might not be optimal for each of the different set of predictions**

✓ **- 1 pts** **You failed to give a better way of choosing the threshold for each model e.g. using a held out validation set**

- **0.5 pts** You included too few or many decimal places in your answer. Between one and two is more appropriate

- **4 pts** You did not answer the question

- **1 pts** Answer too long/answer box resized

4.2 AUC 4 / 4

✓ **- 0 pts** **Correct**

- **2 pts** Your AUC numbers are wrong

- **1 pts** You did not correctly state that the model with the best accuracy does not
have the best AUC score

- **2 pts** You did not identify the reason why alg_4 has a poor accuracy i.e. it is
because 0.5 is a poor choice of threshold of this particular model

- **0.5 pts** You included too many decimal places in your answer. Four or less is more
appropriate, or two or less if you report area in %

- **4 pts** You did not answer the question

- **1 pts** Answer too long/answer box resized

- **2 pts** This is not about overfitting or imbalance, it is about classification thresholds and impact on TPR/FPR.

- **0.5 pts** Algorithm 3 AUC is 6.4%, not 64%.

- **1 pts** Your answer would have been fine if you had not gone on to incorrectly discuss class imbalance -

this is about classification thresholds.

4.3 ROC plots 4 / 6

- **0 pts** Correct
- **2 pts** Your ROC curves do not look like what is expected
- **2 pts** Your ROC curves are not smooth lines i.e. you only created the plot for the thresholded predictions
- **1 pts** You did not plot the ROC curves for all four models.
- **1 pts** You did not plot the four curves on the same plot, making it more difficult to compare them
- **1 pts** Your plot is not clear i.e. you failed to label the axis and to provide a legend
- **1 pts** You failed to describe the performance of alg_3 i.e. it performs much worse than random guessing

✓ **- 2 pts** **You failed to identify that alg_3 can be improved by inverting its predictions**
i.e. a prediction of 0 would become a prediction of 1

- **1 pts** Answer too long/answer box resized
- **6 pts** You did not answer the question

Question 1 : (22 total points) Linear Regression

In this question we will fit linear regression models to data.

(a) (3 points) Describe the main properties of the data, focusing on the size, data ranges, and data types.

Your Answer Here The size of the data is a 50×2 dataset. The first column is revision time, ranging in (2.723 , 48.011). The second column is exam score, ranging in (14.731 , 94.945). All the data are numerical type, float64.

Question 1 continued on next page...

1.1 Data properties 3 / 3

✓ **- 0 pts Correct**

- 1 pts You failed to mention the correct range of the revision time attributes (2.72 to 48.01)

- 1 pts You failed to mention the correct range of the exam score targets (14.73 to 94.94)

- 1 pts You failed to mention the size of the dataset (50 datapoints)

- 1 pts You failed to mention that the attributes are continuous

- 3 pts You did not answer the question

- 1 pts Answer too long/answer box resized

- 0.5 pts Your reported values for the data ranges are incorrect

- 1 pts The range has been specified correctly, but the min and max values have been omitted.

(b) (3 points) Fit a linear model to the data so that we can predict `exam_score` from `revision_time`. Report the estimated model parameters $\mathbf{w}$. Describe what the parameters represent for this 1D data. For this part, you should use the sklearn implementation of Linear Regression.
Hint: By default in sklearn `fit_intercept = True`. *Instead, set* `fit_intercept = False` *and pre-pend* 1 *to each value of* x_i *yourself to create* $\boldsymbol{\phi}(x_i) = [1, x_i]$.

We create $\phi(x_i)$ and use LinearRegression(`fit_intercept = False`) to create the estimated model. It shows the parameter w as `coef_` = [17.89768 1.44114]. The first value is `w_0`, representing the intercept. The second value is `w_1`, representing the slope.

Question 1 continued on next page...

1.2 Linear Model 2.5 / 3

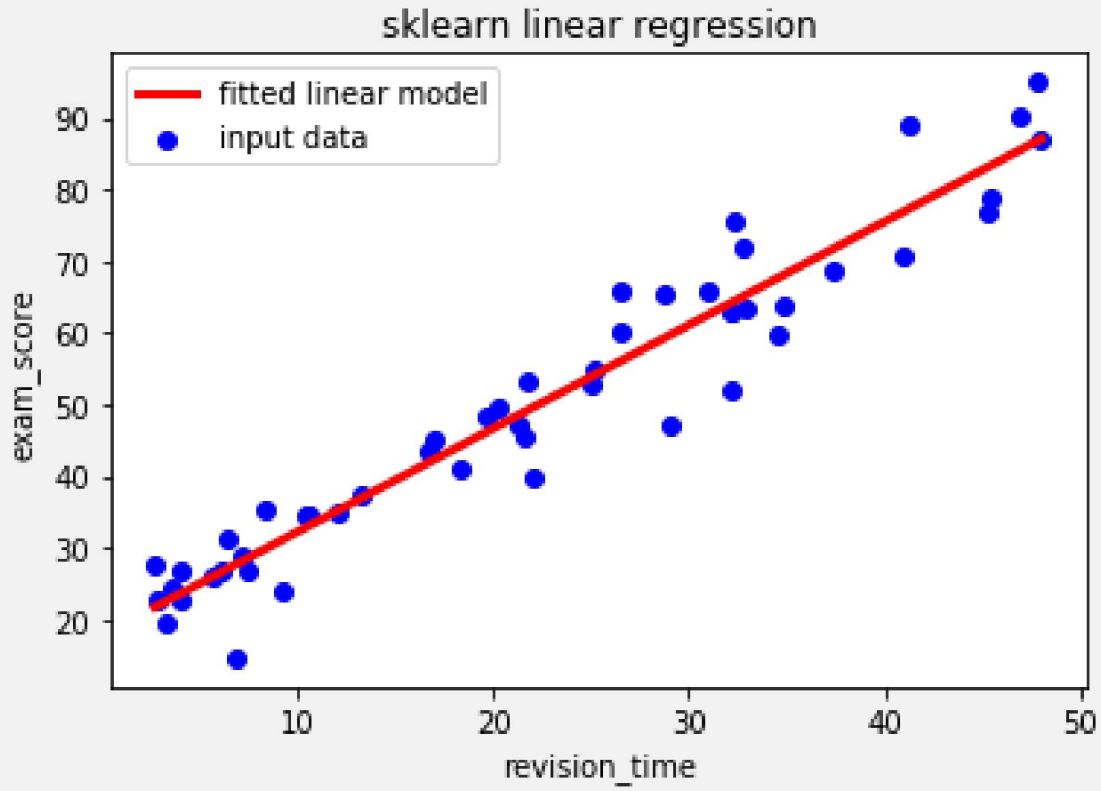
- **0 pts** Correct
- **2 pts** Your reported model parameters are incorrect. They should be 17.898, and 1.441
- **1 pts** You did not explain that the model parameters represented the intercept and slope

✓ - **0.5 pts** **You included too many decimal places in your answer. Four or less is more appropriate**

- **3 pts** You did not answer the question
- **1 pts** Answer too long/answer box resized
- **0.5 pts** Your explanation of the model parameters goes towards the right direction, but is not quite there: you could simply say the model parameters represent the intercept and slope

(c) (3 points) Display the fitted linear model and the input data on the same plot.

This image shows the input data and fitted linear model



Question 1 continued on next page...

1.3 Display 3 / 3

✓ **- 0 pts** **Correct**

- 1 pts Your linear model is a poor fit to the data

- 1 pts The fitted line is not continuous

- 1 pts You did not label the axis

- 0.5 pts You did not add a legend

- 3 pts You did not answer the question

- 1 pts Answer too long/answer box resized

- 1 pts You should not represent your input data using a line

(d) (3 points) Instead of using sklearn, implement the closed-form solution for fitting a linear regression model yourself using numpy array operations. Report your code in the answer box. It should only take a few lines (i.e. <5).
Hint: Only report the relevant lines for estimating **w** *e.g. we do not need to see the data loading code. You can write the code in the answer box directly or paste in an image of it.*

(X and Y here are revision times and exam scores, the same as in solving (b))
The code is as below:

```
phi0 = np.ones(50)[..., None]
X = np.concatenate((phi0, X),1)
coeffs = np.linalg.inv(X.transpose().dot(X)).dot(X.transpose()).dot(Y)
```

output: coeffs = array([17.89768, 1.44114])

Question 1 continued on next page...

1.4 Custom implementation 3 / 3

✓ **- 0 pts** **Correct**

- 3 pts Your code is obviously wrong e.g. you did not perform the pseudo inversion

- 2 pts You did not implement the solution using basic matrix operations e.g. you used np.linalg.lstsq

- 1 pts Your is overly long and does not make use of numpy expressions e.g. not using np.dot or np.linalg.inv

- 3 pts You did not answer the question

- 1 pts Answer too long/answer box resized

- 2 pts You did not implement the solution using basic matrix operations e.g. you used np.linalg.solve

- 2 pts You implemented linear regression only for 1D input attributes.

- 3 pts You fitted a polynomial.

(e) (3 points) Mean Squared Error (MSE) is a common metric used for evaluating the performance of regression models. Write out the expression for MSE and list one of its limitations.
Hint: For notation, you can use y for the ground truth quantity and ŷ (`$\hat{y}$` in latex) in place of the model prediction.

$MSE = \frac{1}{n}\sum_{i=0}^{n=49}(Y_i - \hat{Y}_i)^2$
MSE is sensitive to outliers as it is related to basic statistical concepts, mean and variance.

Question 1 continued on next page...

1.5 MSE 3 / 3

✓ **- 0 pts** **Correct**

- 2 pts Missing square term

- 1 pts y-hat and/or y should have a consistent lowerscript

- 2 pts You added a square root term

- 1 pts Missing a limitation e.g. distorted by outliers or not in the same units as the data

- 1 pts You did not use the suggested notation for the ground truth and model prediction i.e. $$y$$ and $$\hat{y}$$

- 3 pts You did not answer the question

- 1 pts Answer too long/answer box resized

- 2 pts Missing $$\frac{1}{N}$$ term

(f) (3 points) Our next step will be to evaluate the performance of the fitted models using Mean Squared Error (MSE). Report the MSE of the data in `regression_part1.csv` for your prediction of `exam_score`. You should report the MSE for the linear model fitted using sklearn and the model resulting from your closed-form solution. Comment on any differences in their performance.

The MSE for sklearn method is 30.98547.
The MSE for closed-form solution is 30.98547.
They are nearly the same, but the second MSE is about $1.0e^{-}14$ smaller than the first one.

Question 1 continued on next page...

1.6 MSE 2 2 / 3

- **0 pts** Correct

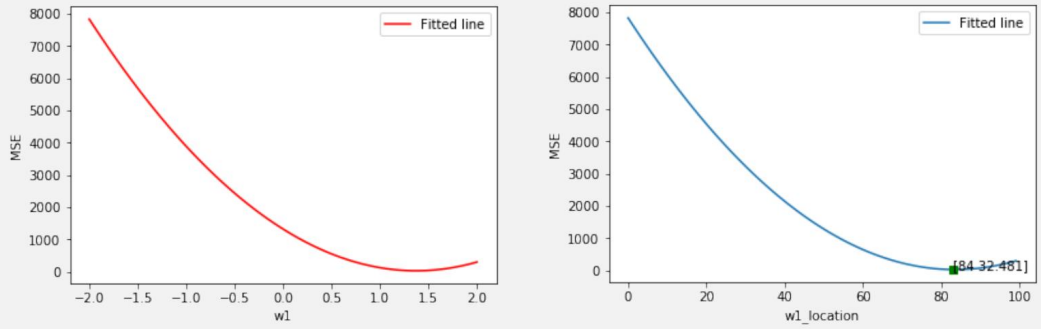
✓ **- 0.5 pts** **You could comment on why MSEs are the same/different**

✓ **- 0.5 pts** **You included too many decimal places in your answer. Four or less is more appropriate**

- **1 pts** You MSE values is wrong. It should be 30.985
- **1 pts** For having different MSE scores for both approaches. They should be very similar for the most significant decimal places
- **3 pts** You did not answer the question
- **1 pts** Answer too long/answer box resized

(g) (4 points) Assume that the optimal value of w_0 is 20, it is not but let's assume so for now. Create a plot where you vary w_1 from -2 to $+2$ on the horizontal axis, and report the Mean Squared Error on the vertical axis for each setting of $\mathbf{w} = [w_0, w_1]$ across the dataset. Describe the resulting plot. Where is its minimum? Is this value to be expected? *Hint: You can try 100 values of* w_1 *i.e.* `w1 = np.linspace(-2,2, 100)`.

The left one is w1-MSE plot, and the right one shows the minimum of MSE and the location.



We find the minimum of MSE is approximately 32.481, under the 84th w_1 which is 1.354.
This value is expected. From previous two methods for creating fitted models, we find the w_1 is about 1.44114, with $w_0 = 17.89768$. Hence, we find that it is reasonable for w_1 to decrease slightly by 0.1 and w_0 increase by 2, since the 'revision time' of this data are positive values ranging from 0 to 50.

1.7 Analysis 3 / 4

- **0 pts** Correct
- **2 pts** You failed to include the plot.
- **1 pts** The range of MSE is incorrect (should be around 8000 for $$w_1 = -2.0$$)
- **2 pts** The plot does not match the correct one in some aspect of the data
- **1 pts** The plot is unclear or insufficiently labelled i.e. you did not label the axis.
- **1 pts** You failed to identify the minimum value of MSE being close to 32.48 or the corresponding $$w_1$$ being close to 1.35

✓ **- 1 pts** **You failed to mention that the plot was convex/that there is a single minimum rather than multiple minima.**

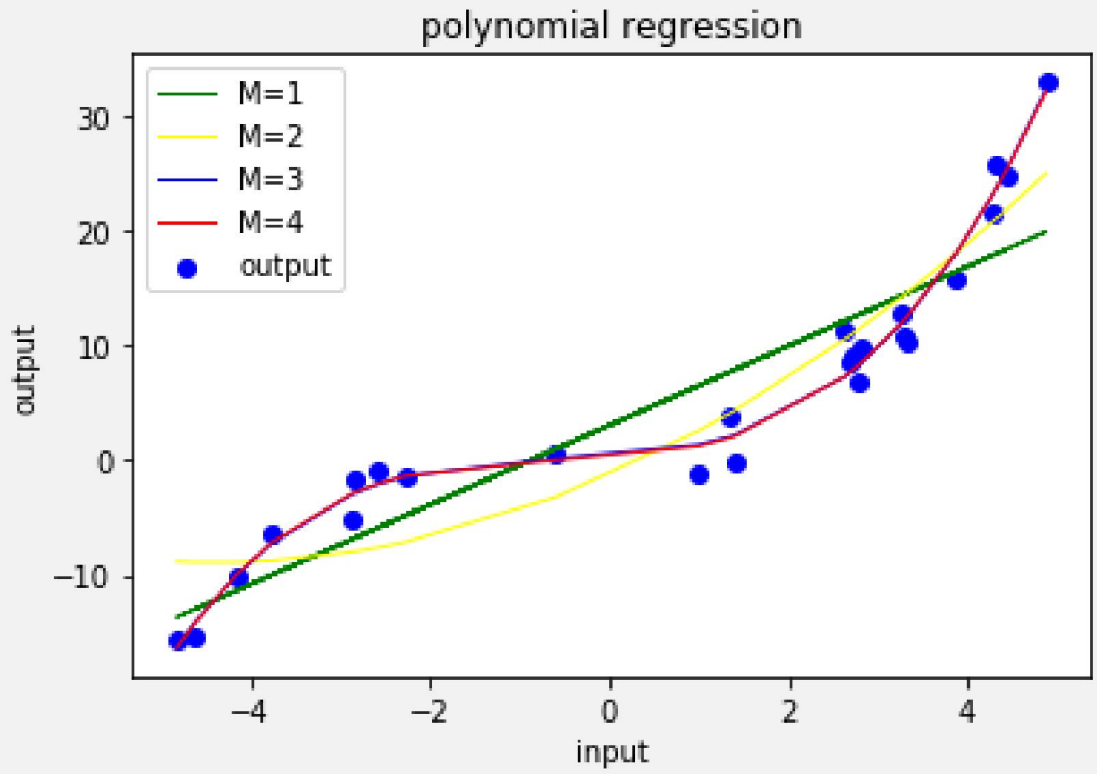
- **0.5 pts** You have identified the plot as concave rather than convex.
- **1 pts** You failed to mention that the resulting minimum $$w_1$$ value was to be expected to be similar to one we found earlier (i.e. close to 1.4)/otherwise refer to the previously calculated result of $$w_1$$ in your argument.
- **0.5 pts** You included too many decimal places in your answer. Four or fewer is more appropriate.
- **1 pts** Answer too long/answer box resized
- **4 pts** You did not answer the question

Question 2 : (18 total points) Nonlinear Regression

In this question we will tackle regression using basis functions.

(a) (5 points) Fit four different polynomial regression models to the data by varying the degree of polynomial features used i.e. $M = 1$ to 4. For example, $M = 3$ means that $\boldsymbol{\phi}(x_i) = [1, x_i, x_i^2, x_i^3]$. Plot the resulting models on the same plot and also include the input data.
Hint: You can again use the sklearn implementation of Linear Regression and you can also use PolynomialFeatures to generate the polynomial features. Again, set `fit_intercept = False`.

This image shows the resulting models and input data



Question 2 continued on next page...

2.1 Polynomial regression 5 / 5

✓ **- 0 pts** **Correct**

- 2 pts Your model predictions look very dfferent from the expected answer

- 1 pts The plots for M = 3 and M = 4 are very different. In this example, they should look almost identical.

- 2 pts You did not plot a continuous line for the models

- 1 pts You did not plot the input data

- 1 pts Your plot is unclear e.g. missing/wrong axes or a legend

- 5 pts You did not answer the question

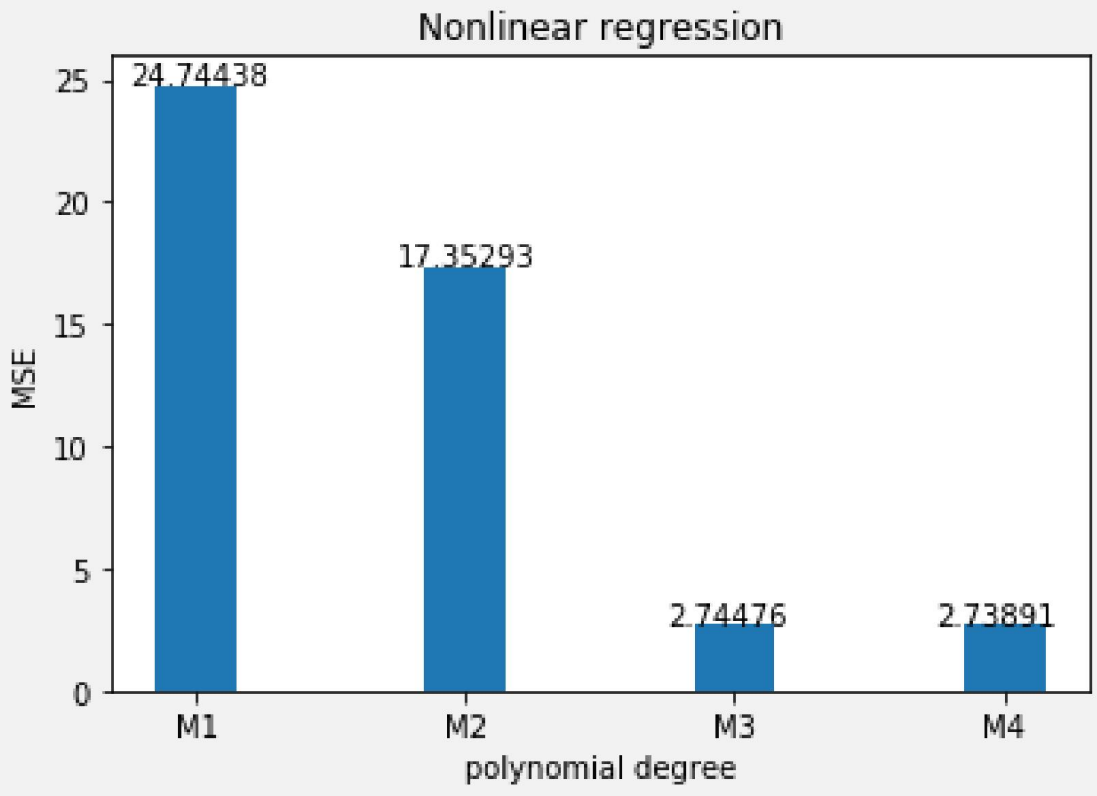
- 1 pts Answer too long/answer box resized

- 0.5 pts The input data should be plotted as points as there is no connection among them

- 3 pts You did not plot some of the lines for the models

(b) (3 points) Create a bar plot where you display the Mean Squared Error of each of the four different polynomial regression models from the previous question.

This bar plot shows MSE of each previous model



Question 2 continued on next page...

2.2 Bar plot 3 / 3

✓ **- 0 pts Correct**

- 2 pts Your bar plot looks very different from the expected answer e.g. M = 3 and M = 4 should have very similar values for MSE, but both are less than M = 1 or M = 2

- 1 pts Your estimated MSE values are incorrect e.g. M = 1 should be around 24.7

- 1 pts Your plot is not clear e.g. missing axes

- 3 pts You did not answer the question

- 1 pts Answer too long/answer box resized

(c) (4 points) Comment on the fit and Mean Squared Error values of the $M = 3$ and $M = 4$ polynomial regression models. Do they result in the same or different performance? Based on these results, which model would you choose?

The fit values for M=3 and M=4 are nearly the same because the previous red line and the blue line basically coincide.
As for the MSE values, MSE of M=3 is 2.74476, and MSE of M=4 is 2.73891.

M	coefficients	MSE
3	[0.40531,0.48562,0.31115,0.19142]	2.74476
4	[0.24293,0.48100,0.35212,0.19153,-0.00169]	2.73891

I would choose M=3 model. Although the MSE values are similar, M=4 model may be overfitting since the coefficient of x_i^4 is too small(-0.00169), which means M=3 is great enough for this polynomial regression.

Question 2 continued on next page...

2.3 Analysis 3 / 4

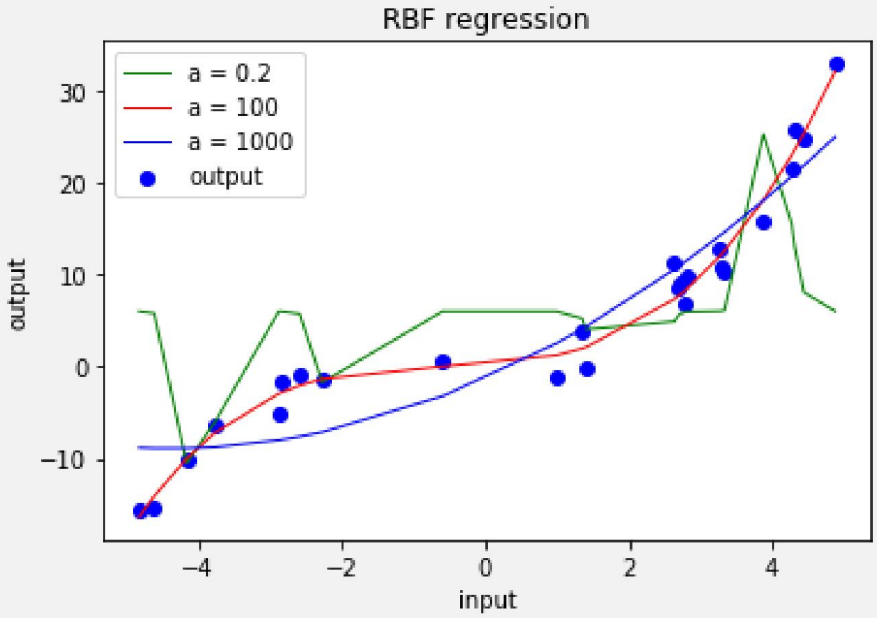
- **0 pts** Correct
- **1 pts** You failed to mention that M = 3 and M = 4 give very similar predictions.
- **2 pts** You said that the M = 4 model is better. The lower parameter M = 3 is a good trade off between complexity and performance i.e. it requires less parameters.
- **1 pts** You did not mention that the weight vector entry corresponding to x 4 i is very small.
- **4 pts** You did not answer the question
- **1 pts** Answer too long/answer box resized

✓ **- 1 pts you did not explain correctly why M = 3 model is better. The lower parameter M = 3 is a good trade off between complexity and performance i.e. it requires less parameters.**

- **2 pts** you did not answer which model to choose. The lower parameter M = 3 is a good trade off between complexity and performance i.e. it requires less parameters.
- **1 pts** you did not explain correctly why M=3 is better. You don't have sufficient evidence to use overfitting as an argument. To comment on overfitting a model should be evaluated on heldout/unseen data. The lower parameter M = 3 is a good trade off between complexity and performance i.e. it requires less parameters.

(d) (6 points) Instead of using polynomial basis functions, in this final part we will use another type of basis function - radial basis functions (RBF). Specifically, we will define $\boldsymbol{\phi}(x_i) = [1, rbf(x_i; c_1, \alpha), rbf(x_i; c_2, \alpha), rbf(x_i; c_3, \alpha), rbf(x_i; c_4, \alpha)]$, where $rbf(x; c, \alpha) = \exp(-0.5(x - c)^2/\alpha^2)$ is an RBF kernel with center c and width α. Note that in this example, we are using the same width α for each RBF, but different centers for each.
Let $c_1 = -4.0$, $c_2 = -2.0$, $c_3 = 2.0$, and $c_4 = 4.0$ and plot the resulting nonlinear predictions using the `regression_part2.csv` dataset for $\alpha \in \{0.2, 100, 1000\}$. You can plot all three results on the same figure. Comment on the impact of larger or smaller values of α.

This image illustrates all three models



α influences the smoothness of model. From this image, we find the line will be smoother as α increases. Hence, if α is smaller, the line is not smooth enough.
In this RBF question, as I find (α=100) is suitable, I print the $\phi(x_i)$ under other models, and find when α is smaller(α=0.2), lots of them are too small(e.g. $e^{-}100$), and when *alpha* is larger(α=1000), most of them are too close to 1(first attribute of $\phi(x_i)$). Therefore, both of them are not great input values and the predicted values will be inaccurate.

2.4 RBF 5 / 6

- **0 pts** Correct
- **3 pts** Your plot looks very different from the expected answer
- **2 pts** You only plotted your model predictions where the input data was. You should have used more input points to better visualize the predictions i.e. to make the plot more continuous
- **1 pts** You failed to mention what would happen if the width parameter is too large i.e. more datapoints further away from the basis center are included resulting in underfittting (or overly smooth predictions)

✓ **- 1 pts You failed to mention what would happen if the width parameter is too small i.e. model predictions will be a constant for points that are not close to the kernel center**

- **1 pts** Your plot is not very clear e.g. you did not plot the input data, it is missing a legend, axis labels or the plot is not complete
- **6 pts** You did not answer the question
- **1 pts** Answer too long/answer box resized
- **3 pts** You failed to provide the figure
- **2 pts** Some of the plotted graphs are different from the expected answer

Question 3 : (26 total points) Decision Trees

In this question we will train a classifier to predict if a person is smiling or not.

(a) (4 points) Load the data, taking care to separate the target binary class label we want to predict, `smiling`, from the input attributes. Summarise the main properties of both the training and test splits.

We seperate target binary label from each dataset.
As for training split, the data type of target label is int64 due to the binary value(0 or 1). It is a 4800×1 dataset, with 2335 1_s and 2465 0_s, which means 2335 smiling person and 2465 not smiling ones. For the 2D coordinates data(size : 4800×136), the data type is float64.
As for test split, the data type of target label is also int64 due to the binary value(0 or 1). It is a 1200×1 dataset, with 592 1_s and 608 0_s. We can find, both of the splits have more not smiling person than the smiling ones. For the 2D coordinates data(size : 1200×136), the data type is float64.

Question 3 continued on next page...

3.1 Dataset analysis 4 / 4

✓ **- 0 pts Correct**

- 1 pts You reported the wrong number of attributes. It should have been 136

- 0.5 pts You did not report the correct train set size (4800)

- 0.5 pts You did not report the correct test set size (1200)

- 0.5 pts You did not report the correct number of positive labels in the train set (2335)

- 0.5 pts You did not report the correct number of negative labels in the train set (2465)

- 0.5 pts You did not report the correct number of positive labels in the test set (592)

- 0.5 pts You did not report the correct number of negative labels in the test set (608)

- 4 pts You did not answer the question

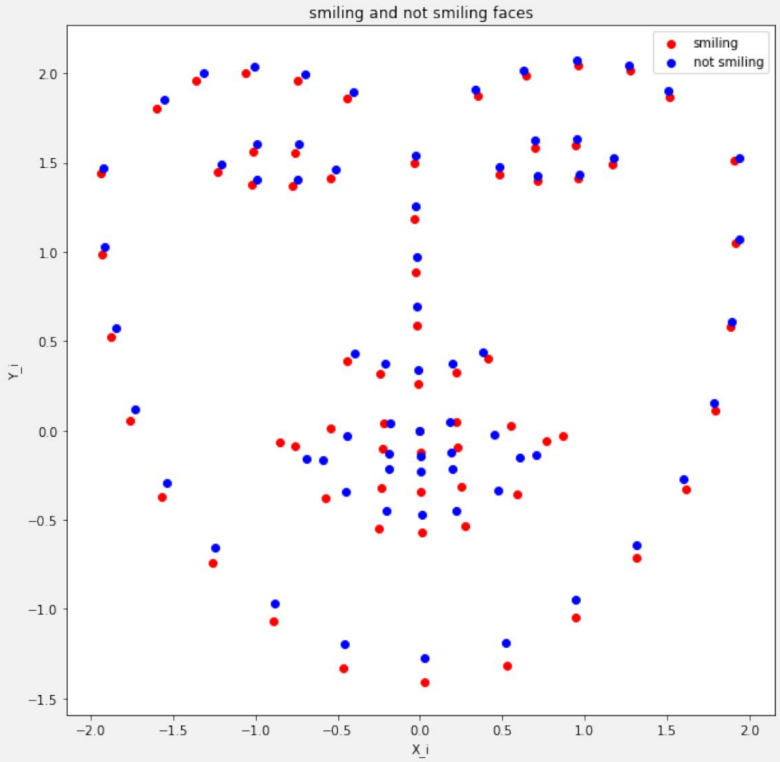
- 1 pts Answer too long/answer box resized

- 1 pts You only reported the ratio of smiling faces to total faces, not the total number of smiling and not smiling faces.

- 1 pts You only reported the ratio of smiling faces to not smiling faces, not the total number of smiling and not smiling faces.

(b) (4 points) Even though the input attributes are high dimensional, they actually consist of a set of 2D coordinates representing points on the faces of each person in the dataset. Create a scatter plot of the average location for each 2D coordinate. One for (i) smiling and (ii) one not smiling faces. For instance, in the case of smiling faces, you would average each of the rows where `smiling = 1`. You can plot both on the same figure, but use different colors for each of the two cases. Comment on any difference you notice between the two sets of points.
Hint: Your plot should contain two faces.

This image shows smiling and not smiling cases



The main differences are about the mouth. We find that the locations of the four feature points at the corners of the mouth of the smiling and not smiling people were significantly different, the locations of the eight feature points on the lower lip were also quite different, and these 12 feature points of the not smiling people were relatively more clustered.

Question 3 continued on next page...

3.2 Analysis 4 / 4

✓ **- 0 pts** **Correct**

- 2 pts You failed to mention any difference between the two sets of points e.g. the mouth is wider and the chin is lower on the smiling face

- 1 pts You did not mention the difference clearly

- 3 pts The plot does not contain two faces

- 1 pts You did not plot both faces on the same figure. This makes it difficult to see the differences

- 1 pts You did not include a legend or way to indicate which points correspond to the train set and which correspond to the test

- 1 pts You plot is missing axes labels, you plotted lines instead of points, etc.

- 4 pts You did not answer the question

- 1 pts Answer too long/answer box resized

(c) (2 points) There are different measures that can be used in decision trees when evaluating the quality of a split. What measure of purity at a node does the DecisionTreeClassifier in sklearn use for classification by default? What is the advantage, if any, of using this measure compared to entropy?

The gini score is a metric that quantifies the purity of the node.
A gini score greater than zero implies that samples contained within that node belong to different classes. A gini score of zero means that the node is pure, that within that node only a single class of samples exist. Therefore, we know that the samples contained within the root node belong to different classes.
Both gini and entropy are measures of impurity of a node. However, Gini is intended for continuous attributes and "Gini" could minimize misclassification.

Question 3 continued on next page...

3.3 Decision Trees 1 / 2

- **0 pts** Correct
- **1 pts** You did not specify the correct measure used by sklearn by default i.e. gini

✓ **- 1 pts** **You did not specify an advantage of gini over entropy e.g. computing entropy requires more computation as you need to take logarithms**

- **2 pts** You did not answer the question
- **1 pts** Answer too long/answer box resized

Incorrect, gini and entropy are equivalent in performance (and misclassification). But one advantage is that gini is more computationally efficient as it does not need log computations.

(d) (3 points) One of the hyper-parameters of a decision tree classifier is the maximum depth of the tree. What impact does smaller or larger values of this parameter have? Give one potential problem for small values and two for large values.

Both of them are not for good models, and will cause loss of accuracy.
Smaller depth of the tree more are the chances of bias tree(underfitting).It does not divide up the data into very distinctive groups as the tree is too shallow.
Larger depth of the tree more are the chances of variance(overfitting). It matches the training data almost perfectly, but does poorly in validation and other new data. Additionally, it implies that some computing time is wasted on achieving overfitting solutions.

Question 3 continued on next page...

3.4 DT Depth 3 / 3

✓ **- 0 pts** **Correct**

- 1 pts You did not mention what happens when you use a maxiumum depth that is too small e.g. underfitting

- 2 pts You did not give two examples of what happens when you use a maxiumum depth that is too large e.g. overfitting, large trees requiring larger storage, and can be slow to evaluate

- 1 pts You only gave one example of what happens when you use a maxiumum depth that is too large e.g. overfitting, large trees requiring larger storage, and can be slow to evaluate

- 3 pts You did not answer the question

- 1 pts Answer too long/answer box resized

(e) (6 points) Train three different decision tree classifiers with a maximum depth of 2, 8, and 20 respectively. Report the maximum depth, the training accuracy (in %), and the test accuracy (in %) for each of the three trees. Comment on which model is best and why it is best.
Hint: Set `random_state = 2001` *and use the* `predict()` *method of the DecisionTreeClassifier so that you do not need to set a threshold on the output predictions. You can set the maximum depth of the decision tree using the* `max_depth` *hyper-parameter.*

Results are presented in the table below .

depth	training accuracy	test accuracy
2	0.795	0.782
8	0.934	0.841
20	1.000	0.816

The second model is best(depth = 8). Its training accuracy is over 0.93 and test accuracy is the largest(0.841). Although the third model matches training data totally accurately, it does poorly in test data(overfitting), as the prediction for test data is more important.

Question 3 continued on next page...

3.5 Hyperparameter tuning 6 / 6

✓ **- 0 pts Correct**

- 2 pts Your reported accuracy numbers are significantly different from what is expected. Are you sure you used the correct random seed and the correct version of sklearn?

- 1 pts You did not report the train set accuracy

- 1 pts You did not report the test set accuracy

- 2 pts You did not correctly identify why max_depth = 8 is the better model i.e. you did not mention the overfitting that happens in the case of max_depth = 20 which clarifies why max_depth = 8 is best.

- 1 pts You reported results with too many digits after the decimal place. Less than three would have been sufficient.

- 6 pts You did not answer the question

- 1 pts Answer too long/answer box resized

(f) (5 points) Report the names of the top three most important attributes, in order of importance, according to the Gini importance from DecisionTreeClassifier. Does the one with the highest importance make sense in the context of this classification task?
Hint: Use the trained model with `max_depth = 8` *and again set* `random_state = 2001`.

Results are presented in the table below .

importance order	1	2	3
name	x50	y48	y29
value	0.131	0.036	0.035

No. Since x50 with average value -0.221, we find the corresponding y50 is 0.040, which is very close to this feature location of not smiling people(-0.181,0.040).

Question 3 continued on next page...

3.6 Attribute importance 3 / 5

- **0 pts** Correct
- **2 pts** Your reported attributes are different from the expected answer of \`x50' \`y48', and \`y29'
- **1 pts** The order of your attributes is incorrect. It should be \`x50' \`y48', and \`y29'.
- **2 pts** You reported the indices of the attributes and not their names i.e. 100, 97, and 59 instead of \`x50' \`y48', and \`y29'.

✓ **- 2 pts** **You did not give a reason why the attributes were likely to be good choice in the context of the task. The most important attribute, \`x50', corresponds to the upper right lip. This is a part of the face that is likely to move a lot when someone smiles.**

- **5 pts** You did not answer the question
- **1 pts** Answer too long/answer box resized
- **1 pts** You answered that the most important attribute does not make sense. \`x50' corresponds to the upper right lip and it is a part of the face that is likely to move a lot when someone smiles.
- **1 pts** You reported one incorrect attribute. The expected answer was \`x50' \`y48', and \`y29'.

(g) (2 points) Are there any limitations of the current choice of input attributes used i.e. 2D point locations? If so, name one.

I think one of limitations is 'not accurate enough'. Any feature of face should be 3D coordinates attribute, such as (x0,y0,z0), since only 2D projection is not enough. When people laugh, height of the side of the lips will also change.

3.7 Analysis 2 / 2

✓ **- 0 pts** **Correct**

- 2 pts You did not give a sensible limitation of the choice of input feature encodings e.g. they are based on absolute pixel locations, it might be better to use relative distances between points or they are subject to noise if detected poorly.

- 1 pts In your answer, you incorrectly said that the data only contains frontal views of faces, it actually contains side views as well

- 2 pts You did not answer the question

- 1 pts Answer too long/answer box resized

Question 4 : (14 total points) Evaluating Binary Classifiers

In this question we will perform performance evaluation of binary classifiers.

(a) (4 points) Report the classification accuracy (in %) for each of the four different models using the `gt` attribute as the ground truth class labels. Use a threshold of >= 0.5 to convert the continuous classifier outputs into binary predictions. Which model is the best according to this metric? What, if any, are the limitations of the above method for computing accuracy and how would you improve it without changing the metric used?

Results are presented in the table below .

	alg_1	alg_2	alg_3	alg_4
classification accuracy	61.6%	55.0%	32.1%	32.9%

The `alg_1` is the best model since its classification accuracy is the largest.
The limitation is the accuracy is a basic evaluation metric that depends on threshold. In the case, we use a threshold>=0.5, and it is not accurate enough to determine the best model only depending on that.
As for the improvement, we can select more thresholds and calculate the accuracy of all models under the corresponding threshold. After that, we can find which model mostly is the best one.

Question 4 continued on next page...

4.1 Classification accuracy 2 / 4

- **0 pts** Correct
- **2 pts** Your accuracy numbers are incorrect (or missing)
- **1 pts** You failed to state that alg_1 has the best performance
- **1 pts** You reported numbers in the range of 0 to 1. You should have used 0 to 100 as the questions asked for %

✓ **- 1 pts Your answer is too generic or you failed to mention a limitation of using a FIXED threshold i.e. the threshold might not be optimal for each of the different set of predictions**

✓ **- 1 pts You failed to give a better way of choosing the threshold for each model e.g. using a held out validation set**

- **0.5 pts** You included too few or many decimal places in your answer. Between one and two is more appropriate
- **4 pts** You did not answer the question
- **1 pts** Answer too long/answer box resized

(b) (4 points) Instead of using classification accuracy, report the Area Under the ROC Curve (AUC) for each model. Does the model with the best AUC also have the best accuracy? If not, why not?
Hint: You can use the roc_auc_score function from sklearn.

Results are presented in the table below .

	alg1	alg2	alg3	alg4
AUC	0.732	0.632	0.064	0.847

No. `alg_4` has the best AUC but the accuracy is just `32.9%`.
I think the main reason is - AUC measures the performance of a binary classifier averaged across all possible decision thresholds, but in (a) we just set $threshold = 0.5$ and calculate the accuracy under that.

Question 4 continued on next page...

4.2 AUC 4 / 4

✓ **- 0 pts Correct**

- 2 pts Your AUC numbers are wrong

- 1 pts You did not correctly state that the model with the best accuracy does not have the best AUC score

- 2 pts You did not identify the reason why alg_4 has a poor accuracy i.e. it is because 0.5 is a poor choice of threshold of this particular model

- 0.5 pts You included too many decimal places in your answer. Four or less is more appropriate, or two or less if you report area in %

- 4 pts You did not answer the question

- 1 pts Answer too long/answer box resized

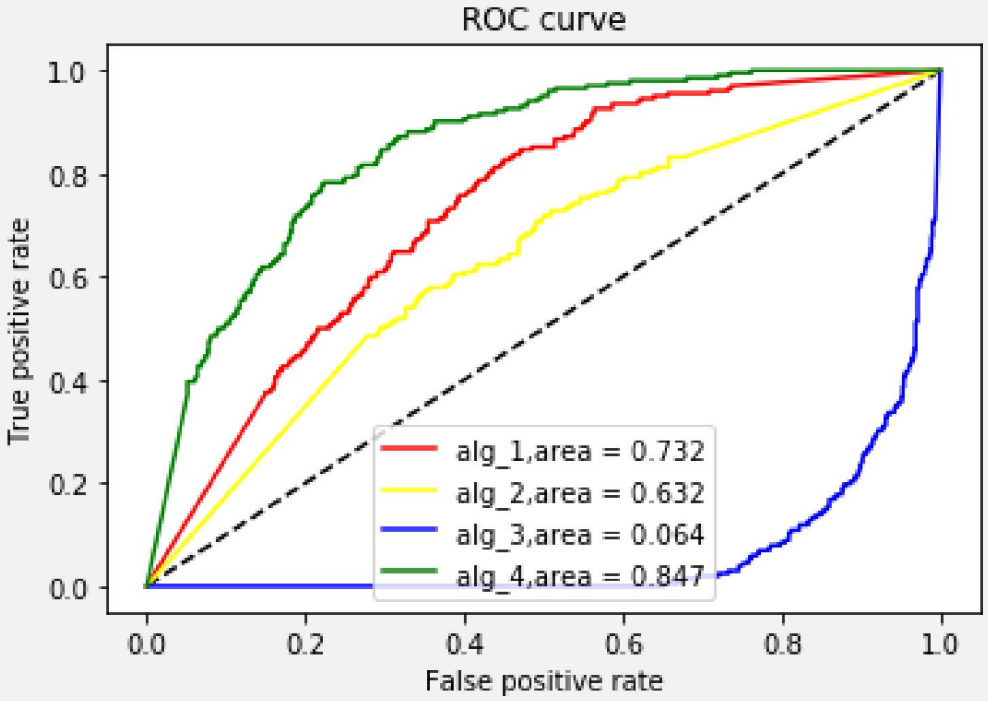
- 2 pts This is not about overfitting or imbalance, it is about classification thresholds and impact on TPR/FPR.

- 0.5 pts Algorithm 3 AUC is 6.4%, not 64%.

- 1 pts Your answer would have been fine if you had not gone on to incorrectly discuss class imbalance - this is about classification thresholds.

(c) (6 points) Plot ROC curves for each of the four models on the same plot. Comment on the ROC curve for `alg_3`? Is there anything that can be done to improve the performance of `alg_3` without having to retrain the model?
Hint: You can use the roc_curve function from sklearn.

This image shows ROC curves for each model



ROC curve for `alg_3` is special, which is the only one under the line 'TP = FP'. Its AUC is just 0.064 and FPR is always greater than TPR.
After considering the definition of each value, if the line is closer to the upper left corner of the plane, its predicted result is better. Hence, as we want to improve the peformance of `alg_3`, we can firstly use $1-TPR, 1-FPR$ to get FNR, TNR, and then use the opposite results for previous prediction because of mirror symmetry.

4.3 ROC plots 4 / 6

- **0 pts** Correct
- **2 pts** Your ROC curves do not look like what is expected
- **2 pts** Your ROC curves are not smooth lines i.e. you only created the plot for the thresholded predictions
- **1 pts** You did not plot the ROC curves for all four models.
- **1 pts** You did not plot the four curves on the same plot, making it more difficult to compare them
- **1 pts** Your plot is not clear i.e. you failed to label the axis and to provide a legend
- **1 pts** You failed to describe the performance of alg_3 i.e. it performs much worse than random guessing

✓ **- 2 pts You failed to identify that alg_3 can be improved by inverting its predictions i.e. a prediction of 0 would become a prediction of 1**

- **1 pts** Answer too long/answer box resized
- **6 pts** You did not answer the question