Machine Learning in Medicine and Biology – Homework 2

Violet Park

September 2025

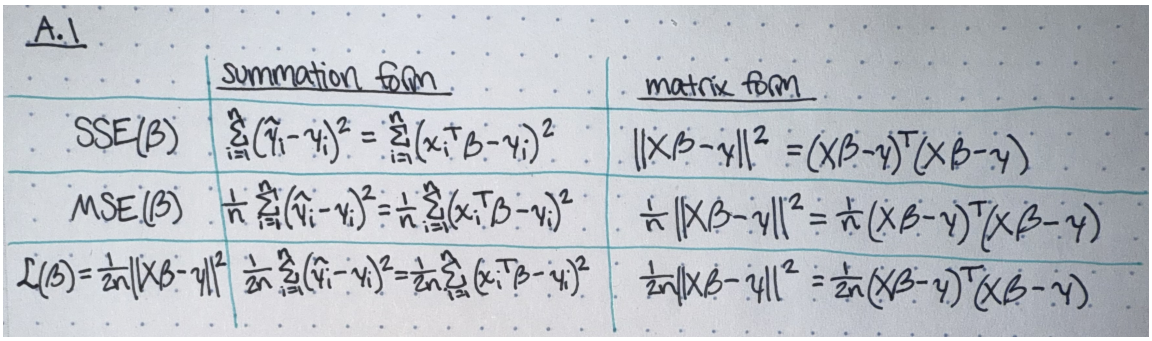
Implement Gradient Descent for Linear Regression

(R or Python)

Beneath the questions, there is attached supplemental theory that will inform your answers. It's mostly things you've seen before but presented here for clarity. It will also define all the math terms you see in parts A and B, and tell you what the answers are.

Part A – Math (show your work) 3pts

A.1 Write SSE(β), MSE(β), and L(β) = 1/2n‖Xβ – y‖2 in both summation and matrix forms.



A.2 Explain in one paragraph why scaling the loss by a positive constant does not change the minimizer. Think in terms of the chain-rule. You don't have to formally derive it.

Scaling the loss by a positive constant does not change the minimizer because it only affects the magnitude of the loss, not its shape or the location of the minimum. In terms of the chain rule, constants are pulled out of the derivative and simply scale the gradient by the same factor, which doesn't change where the derivative equals zero. Essentially, the gradient of the loss before and after scaling will be zero at the same β because the function is only rescaled vertically along the y-axis.

A.3 Look closely at the Jacobian definition. ∂L/∂β = 1/n X⊤(Xβ – y) The rate of change of all of our β's is what?

The rate of change of all of our β's is the gradient/gradient of scalar loss L(β), which is a column of the partial derivatives of all β's.

A.4 Show that setting the gradient ∂L/∂β = 1/n X⊤(Xβ – y) to zero yields X⊤Xβ = X⊤y.

A.4

$$\frac{\partial \mathcal{L}}{\partial \beta} = \frac{1}{n} X^T (X\beta - y)$$

$$\frac{1}{n} X^T (X\beta - y) = 0 \quad (n)$$

$$X^T (X\beta - y) = 0$$

$$X^T X\beta - X^T y = 0$$

$$\boxed{X^T X\beta = X^T y}$$

Part C – Classification Methods Comparison Exercise 3pts

For this section, all you have to do is run the attached code. It is in python. There is an at- tached jupyter notebook in google colab to execute. Save the colab notebook to your google drive. You have to set up a huggingface account https://huggingface.co/, navigate into the dataset https://huggingface.co/datasets/Falah/skin-cancer and click "use this dataset" button on the right. Click on your user avatar on the top-right and click "Access Tokens". Create a new token with one check-mark for read access to gated repositories. Return to your colab session and add the key you just generated to your colab secrets by clicking the key icon on the left of the screen. Give it the name huggingface. Run all the colab cells and read the contents. If you are curious, to run the ViT activate the T4 TPU in colab or pay for the A100, it's not necessary to do so for the assignment.

C.1 Which feature extraction method produced the best results for each classifier? Why do you think this is the case?

For the Naive Bayes classifier, the GLCM feature extraction method produced the best results with an accuracy of 0.75. I think this is the case due to GLCM capturing texture information through the spatial relationship of pixels. This feature extraction method works well for the Naive Bayes classifier because the low number of texture features (5) align well with the Naive Bayes assumption of feature independence.

For the Logistic Regression classifier, the PCA feature extraction method produced the best results with an accuracy of 0.8106. I think this is the case because the principle components of PCA are linearly uncorrelated (no multicollinearity) due to dimension reduction of the features data, and therefore work well with Logistic Regression's linear decision boundary. There is also noise reduction, which benefits the Logistic Regression, since it has a potential to overfit otherwise.

In terms of the histogram feature extraction method, this did not perform that well across either of the classifiers. This may be due to the fact that histograms don't consider spatial/complex relationships within the data and only reflect intensity distributions, which in this case are not enough to accurately classify the images.

The LDA classifier, which does not include feature extraction, had an accuracy of 0.7708. I was also not able to run the ViT classifier due to RAM constraints, so this was left out here.

C.2 Compare the performance of Naive Bayes, Logistic Regression, and LDA. Which model performed best and under what conditions?

The best Naive Bayes performance occurred when using GLCM feature extraction, with an accuracy of 0.75. It also performed fairly close in accuracy (0.7083) when using PCA

feature extraction.

The best Logistic Regression performance occurred when using PCA feature extraction, with an accuracy of 0.8106. It also performed fairly close in accuracy (0.7519) when using GLCM feature extraction.

The LDA model without feature extraction performed with an accuracy of 0.7708.

Therefore, the Logistic Regression model performed the best under the conditions of PCA feature extraction.

C.3 How does LDA perform on the raw pixel data compared to the other models using extracted features? Explain why this might be a poor strategy for images.

LDA performs with an accuracy of 0.7708 using the raw pixel data, compared to the other models which receive accuracies of 0.75 (Naive Bayes) and 0.8106 (Logistic Regression) from using feature extraction. LDA might be a poor strategy for classifying images because it treats each pixel as an independent feature and ignores any potential complex relationships within the data (linear model). It also has the potential of overfitting because the feature number is far greater than the sample number. This would be a poor strategy for even larger or more complex images, exacerbating overfitting and its inability to capture nonlinear decision boundaries.

C.4 The ViT model is a complex deep learning model. What are the advantages and disadvantages of using it solely as a feature extractor for a simple model like Naive Bayes?

The advantages of using ViT as a feature extractor are that it is able to distinguish features in extreme depth from very complex images due to previously learned feature representations, allowing simple models like Naive Bayes to achieve good performance even if the dataset is fairly small. Using a simple model after ViT feature extraction makes the final decision-making process more interpretable and less prone to overfitting.

The disadvantages of using ViT as a feature extractor for simple models are that its features violate the Naive Bayes assumption of feature independence, since the ViT outputs are highly correlated, adn therefore worsen the model's performance. Using ViT solely as a feature extractor also means that it is relying on past training from general images that may not be fine-tuned to the specific cancer image type present within the data. This might lead to limited accuracy for this particular classification task.

C.5 Summarize your findings on the relationship between feature engineering and model performance for this image classification task.

Since image classification is an inherently complex task, it makes sense that feature engineering has a significant impact on model performance for this image classification task. It seems as though feature engineering quality matters more than classifier choice, e.g. Logistic Regression with PCA (accuracy of 0.8106) outperforms Logistic Regression with histogram feature extraction (accuracy of 0.6307). Across all classifiers, it also seems that feature engineering that captures more task-relevant information, like GLCM capturing specific texture properties, perform better than methods such as histogram feature extraction that only capture intensity distributions. In the context of this skin cancer classification task, this makes sense because texture and spatial patterns within the images are very important for diagnosis. Dimensionality reduction seems to be a key process in feature engineering due to its noise reduction and focus on variance, greatly improving model performance. However, it is important to note the classifier's assumptions when choosing a feature engineering method. Overall, model performance degrades without feature engineering that is better fit for the classification task, and feature engineering is more significant in image classification performance than the classifier choice itself.

https://colab.research.google.com/drive/1s-sPmFH2JBPgv_IKXjpwlZP3bMsknA8Q?usp=sharing