Methodology, Approach, & Rationale

1. I imported in the typical libraries that we have been using during the course. This is not an exhaustive list; I imported subsequent libraries later in the notebook.
2. Imported the raw lending club dataset to work with as a data frame.
3. Previewed the first 5 rows to get an idea of what the data frame looks like.
4. Dropped columns which provide no utility i.e., don't tell us anything about the data, or will not add value to our insights.
5. Next, had a look at the null counts for each feature, a lot were significantly high, these are dealt with later in the notebook.
6. Converted revol_util and int_rate into float-type features from object. This would allow me to use the numeric values in the entries in my modelling. I filled imputed the null values by replacing them with 0. This is because if a customer's utilisation or interest rate is 0, it must mean that they do not have any revolving credit accessible to them, and they do not have any debt resulting in an interest rate. I also converted the term feature to a float-type feature.
7. Created a loan default feature by classifying loan defaulters as those applications which were eventually "Charged Off". I intended this feature to be my target variable in the modelling stage.
8. I then encoded the grade column to not only show as a numeric value but be an ordinal feature. In other words, the higher the grade in letter format, the lower the grade in numerical terms. A=0, B=1, C=2, etc.
9. I decided to solely focus on individual type applications and hence removed all joint type applications. I believe these 2 types of applications are somewhat independent of each other, and including joint type applications would adversely affect the quality of my data for modelling and lead to inaccurate insights being drawn from results.
10. Then I identified outliers for several features which I intended to include in my models. My methodology for treating these outliers was to cap them at an upper and lower limit. These limits are defined in algorithm. I did this to maintain a very similar data distribution and reduce the impact of outliers on the performance of my models.
11. I then completed some additional feature engineering by categorising utilisation rates of revolving credit. I thought it might be interesting to see what kind of distribution we had across the different utilisation categories. I decided on 4 general categories, Low, Medium, High, and Very High, each defined in the notebook.
12. I then investigated the purpose feature, attempting to understand where the High and Very High utilisation customers were concentrated, if at all. I was not surprised to see that a good chunk of Very High utilisation customers was concentrated in the 'debt consolidation' and 'credit card' categories. It seems as if they are possibly stuck in this never-ending cycle of taking on more debt to settle previous debt. This is something that the Lending Club should take note of and focus on to manage risk of default. To take this a step further, I could have completed a deep dive into this category of customers,

understand how many are defaulting, is it significant, is it profitable to lend to these individuals?

13. I also encoded and categorised the home_ownership feature to put It into numeric format to include in modelling. Encoding is defined within the notebook. I grouped the smaller categories together given their relative size to the other categories.
14. After this, I produced some summary statistics, to see some statistical metrics of my data frame and each feature.
15. I initialised a baseline model using 10 features. Each justified within the notebook, some of which are features I have engineered myself and included because on my perceived significance of these features.
16. This was a very basic model in which I used a logistic regression model method.
17. As expected, the model performed extremely poorly. It failed to predict any true positive values i.e., any loan defaulters. You can see from the metrics that only the accuracy metric was at an acceptable level. However, this means nothing when the model is predicting all values to be non-defaulters, when in fact there are defaulters within the dataset.
18. Subsequently, I used a resampling method to oversample the minority class (defaulters) to improve the training of my model. This was completed using the SMOTE library, and as you can see from the metrics, accuracy, precision, and recall were all significantly better. Additionally, looking at the confusion matrix, it shows me that the model is now predicting true positives, which is an improvement. However, it is unfortunately predicting a lot of false positives. I hoped to improve this aspect by creating a challenger model using a neural network method.
19. Now, I started dealing with my challenger model. Firstly, I removed all object-type features, and dropped some features which don't tell me a lot about loan defaulters. I also imported the necessary libraries to deal with utilise a neural network model type.
20. I then imputed nulls for numerous features within the data frame. Some of which I replaced nulls with 0, some with the feature's median, and 1 with an extremely large number (999) to act as limit for infinity.
21. I then dropped columns that had null values greater than 75% of the total records. The list of these dropped columns can be found in the notebook.
22. I then selected several features to include in my challenger model. Some of which were used in the baseline model, and additional features I chose because I thought they might have some power of explainability for my target variable 'loan_deafult'.
23. After selecting the features to include in my model, I resampled the data to increase the minority class relative to the population. This was after I had run the challenger model without resampling, and it did not perform too well.
24. Then I scaled my new data frame, so all values were between 0 and 1. This normalisation method helps my model converge more quickly during the training phase, reducing time.
25. I then split my data into training and testing sets.

26. I then initialised my challenger model using a neural network methodology. It was of sequential nature, to allow for a linear stack of layers where each layer's output becomes the input of the next. It consists of 3 hidden layers and an output layer. After building the model, I trained it on the training data split. The hyperparameters used can be seen within the notebook, such as the no. of nodes, epochs, validation split size, etc.
27. After training the data, I then used the model to predict y values ('loan_default'). As you can see from the metrics in the notebook, and the confusion matrix, the model performed extremely well. So well, that I would need to question if the problem of overfitting is present within my model.
28. As you'll see in the subsequent graphs, I have visualised the accuracy, precision, recall, and loss for the training set and validation set. An indicator of overfitting would be the training metrics (blue) increasing while the validation metrics (orange) are either staying stagnant or increasing at a relatively slower pace.

Baseline Model (Raw)	Baseline Model (Resampled – SMOTE)	Challenger Model (Resampled – SMOTE)
Accuracy: 0.872515753756665 Precision: 0.0 Recall: 0.0 F1-Score: 0.0 ROC-AUC: 0.6528444872135901 Confusion Matrix: [[16200 0] [2367 0]]	Accuracy: 0.641869793273681 Precision: 0.6382295199000535 Recall: 0.6600221483942414 F1-Score: 0.6489429271390981 ROC-AUC: 0.7145623635774893 Confusion Matrix: [[10075 6081] [5526 10728]]	Accuracy: 0.9771366862079605 Precision: 0.9720748734833242 Recall: 0.9825588561567854 F1-Score: 0.9772887485824624 ROC-AUC: 0.9972418216639547 Confusion Matrix: [[15726 458] [283 15943]]

29. I have taken a snapshot of the model metrics for each of the 3 attempts, each labelled in the table above. As you can see, utilising the SMOTE resampling method increased the performance of my baseline model drastically by handling the class imbalance and overrepresenting the minority class; customer who have defaulted on their loans. Initially my baseline model was not predicting any true positives and essentially predicted all values to be negative i.e., False. From the statistics in the table, you can also see that my challenger model is performing significantly better than both versions of my baseline model. Utilising a sequential neural network method has proved to be beneficial.
30. On the point of overfitting, I would agree that overfitting is present. However, to what extent is difficult to say. A suggestion for next steps for the Lending Club would be to utilise my model on new application datasets and observe its performance.
31. For purposes of conciseness, I will focus on my challenger model given the poor performance of my raw baseline model, and how much better than the resampled baseline model. The advantages would be the following:
    a. Simplicity in building. It very easy to adjust hyperparameters, as well as develop the model, such as adding more hidden layers, or including dropout layers. I have not included dropout layers as I did not deem it necessary, but this would have helped me reduce any overfitting, whatever level it is.

b. Ease of debugging. Should a user come across an error, it is very easy to identify at which step this error occurred and fix this. For example, when resampling my data, I came across an error where the number of values to be predicted did not align with the number of records in my train/test split. This was an easy fix as I simply had to correct the variable the algorithm was looking for.
c. It is extremely intuitive. A beginner user should be able to grasp the process and what is happening under the hood quite easily. A model is being initialised by passing data through various nodes at subsequent layers and is learning from each iteration, hence becoming more precise and accurate when predicting the target variable.

32. Some disadvantages tied to my challenger model are:
    a. The long training times. If this model was to be used on a larger dataset, which would be the case at a large financial institution, it is hard to say how long it might take to train the model.
    b. Additionally, sequential neural network models are prone to overfitting as observed with my model. Although the accuracy would dictate that overfitting may not be significantly present, the precision and recall metrics provide us with an opposite perspective.
    c. Also, this type of modelling requires careful hyperparameter tuning. Should the number of nodes, number of epochs, or validation split size be sub-optimal (which will more than always be the case), the model could perform extremely poor, and overfitting could become unacceptably prevalent.
33. With regards to deploying this model, we could utilise a proof-of-concept application with uvicorn and streamlit as shown in the course. We could allow for users to add new loan application datasets and append these to the existing dataset. This would be good to understand whether the model will be performing as well as it has been when new data is introduced. It would also be a cost-effective method to test the model in its early stages. With regards to scalability, the Lending Club can always look to utilise the model in the cloud environment. This would of course need to be justified with a business case which establishes its value-add to operations and the firm holistically.
34. It is difficult to say what the impact of this project would be for the Lending Club in reality. Assuming the Lending Club is still a very immature and small lending firm, they may potentially be using very basic statistics to make judgements on loan applications. This introduces a lot of risks around human judgement. Introducing a machine learning model which is more objective would help manage risks. Additionally, one could argue, that a neural network model which picks up on trends invisible to the human eye, could help tackle the risks around adverse selection. The model will slowly learn the characteristics of default applications and utilise this to predict loan defaults for applications where

there is an asymmetry of information; the applicant has more information about their circumstances than the lender.

35. With regards to ROI, I imagine the deployment of this model would very high. The model will assist the Lending Club with making better decisions around lending and reduce the risk of lending to potential defaulters.