1. **Dataset selection**
2. Repositories including UCI and Kaggle
3. Formulate hypothesis and questions in line with the literature
4. Classification or regression
5. Supervised or unsupervised
6. **Data pre-processing** 
   1. Import (reatdable)
   2. Cleaning
7. Change column names
8. Removing unwanted predictors
9. Testing and visualising missing values followed by removing missing values and visualising to confirm
10. Reordering columns so that they are in a meaningful order
    1. Descriptive statistics and visualisation
11. Summary statistics (summary and groupsummary)
12. Plotting correlation matrix
13. Plotting distribution by histograms and boxplots

**2.4** Data normalisation (if of different magnitudes). Binary data don't need normalisation

1. **Dividing dataset into training and test datasets**
2. Identify proportions according to the desired method based on literature (e.g. 0.8 for **training** and 0.2 for **test**)
3. Shuffle data so that orders are random (to avoid bias) using randperm
4. rng(“default) to ensure reproducibility
5. **Machine learning solutions: the first method (e.g logistic regression, random forest)**
   1. Structuring the first method
6. This starts with cross-validation (e.g. logistic regression) or bootstrapping (random forest) on **training** dataset (for the former, K-fold for small and holdout for larger datasets). For consistency of comparison between two methods cross-validation can be used for both methods (?). How to account for using either for two methods for fair comparison; part of discussion?
7. For cross validation, define the number of folds appropriate for the selected method (e.g. kfold = 10). Literature can inform this decision
8. Using cvpartition, the entire **training** dataset is split into k parts of equal sizes (**training** and validation): a random nonstratified partitioning for k-fold cross-validation on n observations
9. Within a for loop, **training** and **validation** sets (xtrain and ytrain, xval and yval) are defined using “test” and “training” functions of cvpartition. With each iteration (if k=10), 9 folds used for **training** and 1 fold for **validation**
10. Model is defined: mdl = modelname(....)
11. Prediction for **validation** set performed
12. Mean squared error (mse) defined. Loop ended
13. Outside of the loop the average mse is computed
14. Instead of mse it might be more appropriate to calculate logistic loss for logistic regression
15. Can choose to select the best performing fold and to run on the entire training dataset. If so, one way to achieve this is to define an if statement within the for loop
    1. Evaluating the performance of best model from above on the entire **training** dataset
16. Calculating precision, recall and F1
17. Plotting confusion matrix to visualise performance
18. Plotting ROC and AUC
19. Discuss the outcomes
    1. Optimisation
20. Explore "predictor importance" in view of feature selection: some steps to achieve this include: plotting correlation coefficient, stepwiseglm, plotSlice(model) and removing less contributory predictors by either "removeTerms" or "step"
21. Adding features by feature engineering
22. Identifying outliers
23. Examining residuals
24. Discuss the outcomes
    1. Modifying the first method
25. Choosing final tuning parameters to improve precision, recall and F1
26. Implement these parameters to build new models
27. Discuss the outcomes
    1. Prediction of the **test** dataset using the improved model(s)
28. xtest and ytest are defined before a prediction
29. Check the performance of the test prediction
30. Discuss the outcomes
31. **Structuring additional second method according to the workflow in step 4**
32. **Comparison of the two methods**
33. Discuss in line with literature: pros and cons of methods
34. Analysis and critical evaluation of results, conclusions and future work
35. Report within a poster

**7. Ensure reproducibility**

1. Record and provide all necessary information for the code to run on City’s lab computers
2. To achieve this, enable VPN, connect via <https://labs.city.ac.uk/> and run your Matlab code