Minimum Viable Product

To ensure that the project that you're aiming for makes some sense it's wise to aim first for a minimum viable product (MVP).

Your MVP should show that it is possible to achieve learning for your dataset. This means you will need to have implemented at least some of your preprocessing, a train-validation-test split, a model and a suitable metric of performance.

Your system should perform **above random guessing**. For classification this is fairly easy to figure out (accuracy > percentage of sample in most common class), but for regression this is not so easy.
If you have a regression task, compare the validation MAE/MSE/RMSE of your predictions against what it would be if you always predict the mean.

Only showing your model validation accuracy/error is not sufficient, it should be compared to the random guessing performance. Note that in a balanced binary classification problem an accuracy of 50.02% is probably not actually better than the random guessing accuracy of 50%. Make sure you use **validation** metrics to assess performance. A model that can only overfit is not better than random guessing.

The assignment does not ask for a user-compatible app/website/application, like normally the case with a "Minimal Viable Product" in commercial software development. **Therefore, model deployment as an API is not required.**

This assignment is pass/fail. You get full points by submitting an MVP above random guessing before the deadline. Otherwise you get no points.
Please submit the following:

1. A short description of what you've already implemented. This helps your TA assess whether things are on track.
2. **Evidence that you achieve above random guessing model performance on your validation data**. This can be your **validation** accuracy vs. random guess accuracy.
3. An outline of what you're still planning to do. This helps your TA assess whether things are on track.
4. **A way for your TA to access your github repo (important!).** This can be a link if your repository is public (recommended), or you can add your TA's github account as a collaborator. **Make sure this actually works. You can also arrange this with your TA before the deadline!**
5. A short statement describing what each team member has contributed.

You don't need to wait until the deadline to submit, and you're certainly free to proceed with your project before the MVP deadline.
Late submissions will receive 0 points. Your submission should not exceed 1 page.