This is the video for the coursework of neural computing. Multilayer perceptrons and support vector machines were compared using the dataset regarding customer churn in a bank. First, an initial analysis of the dataset was conducted, and then the MLP model's baseline was established and hyperparameter tuning was conducted through grid search and manual search to identify the model with the highest test accuracy and select it as the final model. Similarly, after establishing the baseline of the SVM model, each test accuracy was checked through hyperparameter tuning using grid search and manual search, and the model with the highest accuracy was selected as the final model. In this process, SVM required a lot of time for training and grid search, perhaps because it judged that the dataset was relatively large. So, the hyperparameters of the SVM model were tuned through the halving grid search provided by scikit learn. Also, another SVM baseline model using the LinearSVC code for large datasets was built and compared with the model using SVC code. In conclusion, the model that underwent hyperparameter tuning after applying the RBF kernel to the SVC code showed the highest accuracy and that model was selected as the final model. Despite numerous hyperparameter tuning, and changes of the number of iterations and epochs, if you look at the training process of each final model, you can see a slight spike during validation in the MLP's loss plot and SVM's learning curve. Through this, it appears that a more detailed adjustment of the number of epochs is necessary in the final MLP model and additional hyperparameter tuning, such as adjustment of regularization C, is necessary in the SVM model to prevent overfitting. Then, reproducibility was confirmed by loading each final model on this notebook only for testing and using a test set modified for each model. Not only the test accuracy but also by comparing the ROC curve, calculation of AUC, precision-recall curve, and average precision score, the MLP model confirmed that it had better performance than the SVM model for this bank churn dataset. Although SVM took a lot of time and memory size while training and MLP also took a considerable amount of time due to K-fold cross-validation and epochs, if more detailed hyperparameter tuning in regularization and early stopping were implemented in code, a more stable model with better accuracy might be achieved. Thank you.