Week 11 Assignment

Method used	Dataset size	Testing-set predictive performance	Time taken for the model to be fit(seconds)
XGBoost in Python via scikit-learn and 5-fold CV	100	0.900	0.164
	1000	0.95	0.26
	10000	0.97	0.825
	100000	0.987	4.031
	1000000	0.991	43.47
	10000000		
XGBoost in R – direct use of xgboost() with simple cross-validation	100	0.9500	0.07
	1000	0.9700	0.0992
	10000	0.9475	0.111
	100000	0.9499	0.373
	1000000	0.95	2.878
	10000000	0.9503	57.55
XGBoost in R – via caret, with 5-fold CV simple cross-validation	100	0.93	0.120
	1000	0.940	0.300
	10000	0.935	0.890
	100000	0.938	2.98
	1000000	0.940	20.68
	10000000	0.93	90.459

Observation

Predictive Performance:

Python's XGBoost via scikit-learn consistently achieved the highest accuracy (around 0.99 for 1 million samples).

Direct use of xgboost in R achieved slightly lower accuracies, but remained consistent (~0.95) across dataset sizes.

Caret-based training in R achieved slightly lower accuracies compared to direct xgboost use.

Computation Time:

Direct use of xgboost in R was the fastest across all dataset sizes.

Python's scikit-learn was moderate in terms of training time.

Caret in R was the slowest, especially for very large datasets.

Scaling Behavior:

All methods showed increasing training time with larger datasets, as expected.

However, R's direct xgboost method scaled more efficiently compared to Python and caret-based training.

Conclusion.

It took a very long time for me to do this in R . Xgboost-caret in R took almost 3hrs to run. While in the python the entire process was less time taking

If highest predictive accuracy is important , XGBoost via Python (scikit-learn) is recommended.

If speed and computational efficiency are critical, especially for very large datasets, direct use of xgboost in R is preferable.

Using caret in R adds extra overhead and is less efficient compared to the direct xgboost method.

Overall, the choice between Python and R should depend on the specific accuracy requirements and computational resource constraints.