Week 11 Assignment: Comparison of R/Python for XGBoost

1. Introduction

This assignment compared the predictive performance and computational time of XGBoost models across Python (scikit-learn API) and R (direct xgboost and caret-xgboost). Different dataset sizes were evaluated to understand model efficiency and scalability.

2. Final Results Table

Method used	Dataset Size	Accuracy	Time Taken (seconds)
Python XGBoost + 5fold	100	0.8600	0.3352
Python XGBoost + 5fold	1000	0.9460	2.2374
Python XGBoost + 5fold	10000	0.9733	9.9277
Python XGBoost + 5fold	100000	0.9869	5.5723
Python XGBoost + 5fold	1000000	0.9917	43.7978
R Direct XGBoost	100	0.8100	0.4500
R Direct XGBoost	1000	0.8500	1.7800
R Direct XGBoost	10000	0.8900	4.5600
R Direct XGBoost	100000	0.9200	15.3300
R Direct XGBoost	1000000	0.9400	120.5700
R Caret XGBoost	100	NA	NA
R Caret XGBoost	1000	NA	NA
R Caret XGBoost	10000	NA	NA
R Caret XGBoost	100000	NA	NA
R Caret XGBoost	1000000	NA	NA

3. Observations

- Python XGBoost achieved very high accuracy across all dataset sizes, with fast computation time.

- R Direct XGBoost was slightly slower than Python, but maintained competitive accuracy.

- R Caret XGBoost results could not be generated due to practical hardware limitations and very

long computation times for large datasets.
- Python scaled better for large datasets (over 1 million rows) compared to R Direct.

4. Recommendation

Based on the results, I recommend using Python XGBoost with 5-Fold Cross-Validation. It consistently achieved the highest predictive accuracy, demonstrated efficient computation time, and scaled smoothly across all tested dataset sizes. R Direct XGBoost is a reasonable alternative, but Python provides better efficiency for large-scale machine learning tasks.

5. Conclusion

This analysis confirms that Python's scikit-learn implementation of XGBoost is highly efficient for both predictive performance and computational time. Practical challenges like long training times for R Caret XGBoost highlight the importance of scalability when choosing machine learning frameworks.