Unit 4: Assignment

CIS625 – Machine Learning for Business

Najse Foster

Professor JC Martel

M.S. in Information Systems and Business Analytics

Park University

April 16, 2025

SHAP Value Comparison: Random Forest vs XGBoost

The SHAP summary plots showed how each model prioritized features.

In both models, "Survey Question" was the most impactful variable. That made sense. It is the most detailed layer in this dataset and heavily influences the final prediction.

In the Random Forest model, I noticed a wider spread of influence. Other features like "Stratification Detail" and "State Abbreviation" still held weight. This makes sense given how bagging models spread importance across trees. They do not depend on just a few features. They aggregate. That means more features show up as important, though not equally so.

XGBoost was more focused. After "Survey Question," the drop in importance was sharp. It leaned into "Health Category" and almost ignored others. Boosting models like XGBoost build trees sequentially, adjusting to errors. This makes them sharper in targeting a few high-impact variables. It also makes their decision process more narrow.

PDP Differences: Stratification Detail

For the partial dependence plots, I used "Stratification Detail" for both models. This variable reflects demographic splits like age or income and was selected because it showed moderate SHAP influence across both models.

Random Forest showed more consistent variation across values. Its plot reflected gentle climbs and drops, which means the model reacted gradually to differences in the encoded values.

XGBoost’s PDP showed sharper changes. Fewer plateaus. This confirmed the model was more aggressive. It likely ignored many stratification groups and focused on a few that affected the outcome the most.

This type of comparison matters in real-world analysis. If I were reporting this data for compliance review, I might prefer Random Forest for explainability. But if the goal was pure prediction, XGBoost might be more effective.

Technical Roadblocks

Getting the visuals right was a bigger task than expected. At first, none of the SHAP plots worked. The older initjs() method failed, and I had to troubleshoot with shap.Explainer(). Even then, the plots came out empty until I used a sample size of 100 to reduce processing time.

The default feature names were also unusable. Terms like "Stratification1" or "Class" don’t mean much without context. I renamed each column to something that a human could understand. That included terms like "Health Topic," "State Abbreviation," and "Start Year."

Everything now makes sense to someone outside the codebase. That was my goal. I wanted my plots to speak clearly.

References

Microsoft. (2024). View usage data for your SharePoint site.  
<https://support.microsoft.com/en-us/office/view-usage-data-for-your-sharepoint-site-2fa8ddc2-c4b3-4268-8d26-a772dc55779e>

SHAP. (2024). SHAP documentation and explainer examples.  
<https://shap.readthedocs.io/en/latest/example_notebooks/api_examples/plots/bar.html>