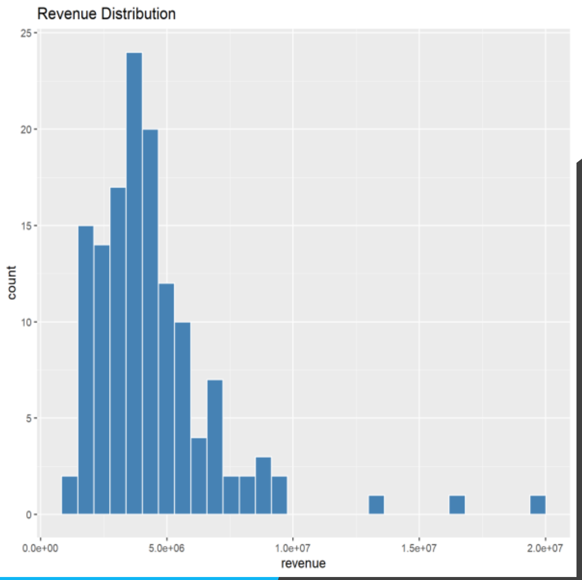
# STAT 348 – Final Project Restaurant Revenue Prediction

*A project by*

Hongting (Ketherine) Wang

# Problem Background

- **Kaggle Regression Competition**
- **Predict annual restaurant revenue from 42 features**
  - **Id:** Restaurant Id
  - **Open Date (Date):** Opening Date for A Restaurant
  - **City (Categorical):** City Name of the Restaurant
  - **City Group (Categorical):** Type of the City
    - Big Cities/Other
  - **Type (Categorical):** Type of the Restaurant
    - DT: Drive Through/FC: Food Court/IL: Inline, MB: Mobile
  - **P1 – P37 (Numeric):** Demographic, Real Estate, and Commercial Data
  - **Revenue:** Target Variable
    - Transformed Revenue of the Restaurant in a given year
- **Very small training set (137 rows) vs. huge test set (100,000 rows)**
- **Include obfuscated operational metrics (P1 – P37) and unseen levels/categories in test set**




# Feature Engineering

- Remove two extreme revenue outliers (revenue > 15,000,000)
- Extracted **Day**, **Month**, **Year**, **Days Open** (today's date – open date) from Open Date
- Converted categorical fields: Type, City Group
- Removed non predictive and redundant fields: Id, City, Open Date (original)
- Preprocessing Recipes
  - Linear Models: Unknown/Novel Handling + Dummy Encoding + Normalization
  - Tree Models: Unknown/Novel Handling + Dummy Encoding

# Model Comparison

### Models Evaluated:

- Elastic Net Regression
- Random Forest (BEST)
- XGBoost

### Kaggle RMSE:

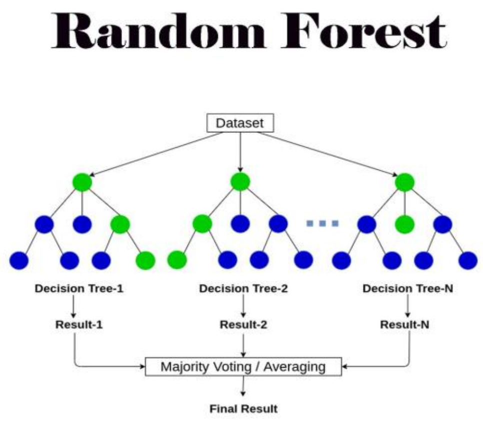
**Project Cutoff: 1,757,539**

- 1,752,060
- 1,613,431
- 1,671,891

### Observations

- Tree-based models outperform linear
- Random Forest is more stable on small datasets
- XGBoost struggled with limited sample size

- # Random Forest Overview
  - Ensemble of Decision Trees
  - Uses Bootstrap Sampling
  - Random Feature Selection at Each Split
  - Predictions Averaged across Trees
  - Naturally Handles Nonlinearities & Interactions
- # Key Hyperparameters Tuned
  - **mtry:** number of variables sampled per split
  - **min_n:** minimum observations in leaf
  - **trees:** 35
- # Why It Outperformed Others
  - Robust to Overfitting on Small Datasets
  - Captures Nonlinear Relationships in P1 – P37
  - Insensitive to Scaling and Outliers




# Random Forest Algorithm

# Final Results & Key Takeaways

| | | |
|---|---|---|
| el_submission.csv<br>Complete (after deadline) · 1h ago | 1842239.45201 | 1752060.20626 |
| rf_submission.csv<br>Complete (after deadline) · 2h ago | 1747449.97117 | 1613431.57091 |
| xgb_submission.csv<br>Complete (after deadline) · 1h ago | 1747189.57391 | 1671891.43337 |

- **Key Takeaways**
  - Feature engineering had major impact
  - Tree methods outperform linear for this dataset
  - Random Forest is most stable with limited data
  - All models met the project cutoff