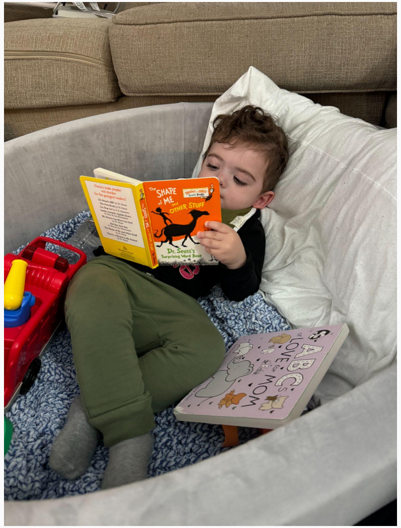
BST 219
Core Principles of Data Science

Lecture 29: Machine Learning continued
December 17, 2024

Recipe of the Day!

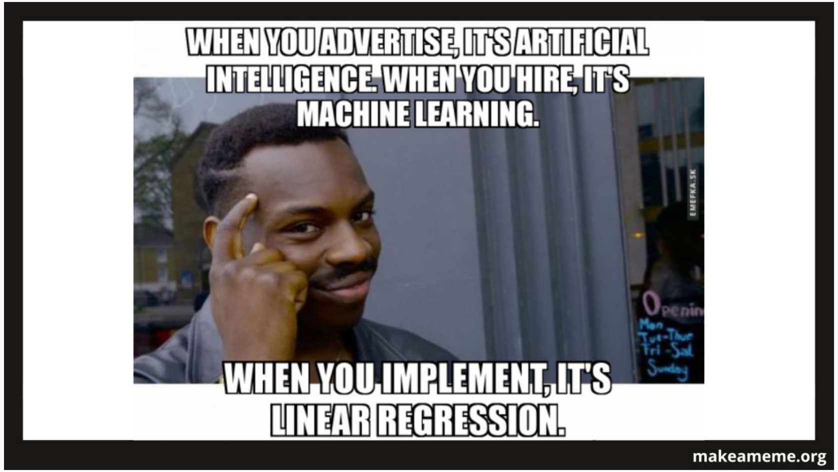
Muddy Buddies



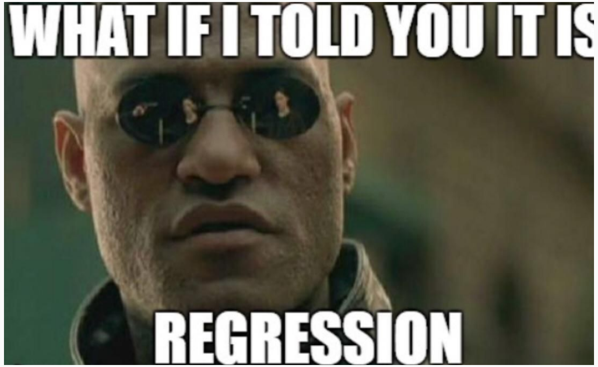
Emre hopes you get time to relax and read over the break

Agenda

- Announcements
  - No lab this week!
  - No office hour on Thursday
  - Please complete the course evaluation - we value your feedback!



- Continue Machine Learning module
  - Answer a couple of questions about Random Forests from lecture last week
  - Regularization



Coding Question of the Day!

Same setup as 12/10 question of the day, but different model.

Using the `gapminder` dataset, fit a **Random Forest** model that predicts fertility (low vs high) using life expectancy, infant mortality, population, and gdp as predictors and data from the year 1989 only. Make sure it is a **bagging** model.

Compare the **accuracy**, **sensitivity**, and **specificity** for this model to the tree model from the December 10th question of the day.

The code that categorizes fertility into low (fertility <= 4) and high (fertility > 4) groups, with 0 indicating low fertility and 1 high fertility, has been provided. The training and test sets using 70% of the data for the training set and 30% for the test set have also been coded for you.

Questions about Random Forests

Do other models use the Gini index?

The Gini index, also known as the **Gini impurity**, is a metric used primarily in **decision tree-based machine learning models**. It measures the likelihood of incorrectly classifying a randomly chosen element in a dataset if the dataset were split according to a particular attribute.

Models That Use the Gini Index:

1. **Decision Trees**
2. **Random Forests**
3. **Gradient-Boosted Trees**:
   - Gradient-boosted models like **XGBoost**, **LightGBM**, and **CatBoost** allow using the Gini index for splitting, although other criteria such as entropy or customized loss functions are also available.

Why Use the Gini Index?

- **Efficiency**: Gini is computationally simpler compared to entropy because it does not involve logarithms.
- **Interpretability**: It provides a clear metric for deciding the best feature to split on at each step of building a tree.

If you're working with other types of models (e.g., linear models, SVMs, neural networks), the Gini index is generally **not** used because those models do not involve splitting nodes based on feature values. Instead, they rely on other optimization techniques like minimizing loss functions or maximizing margins.

Why do Random Forests use bootstrapping instead of cross-validation?

1. Reducing Overfitting Through Ensemble Learning

- Random forests are designed to reduce overfitting by aggregating predictions from multiple decision trees.
- By training each tree on a slightly different dataset (bootstrap samples), the trees become less correlated, leading to a more robust model.
- This diversity improves generalization without needing explicit cross-validation.

2. Efficient Use of "Out-of-Bag" (OOB) Error

- A key advantage of bootstrapping is that it leaves out a portion of the data in each bootstrap sample, typically about **1/3rd of the data**. This is known as the **out-of-bag (OOB) data**.
- These OOB samples act as a built-in test set for the random forest. The model’s performance on OOB data provides an unbiased estimate of the generalization error, eliminating the need for separate cross-validation.
- OOB error reduces computational overhead since you don’t have to split data repeatedly for cross-validation.

Why do Random Forests use bootstrapping instead of cross-validation?

3. Reduced Computational Cost

- Cross-validation involves splitting the data into k-folds and training the model k times (or more for nested CV). This can be computationally expensive for a large dataset.
- In contrast, bootstrapping trains each tree independently and in parallel, making it more efficient, especially for large datasets or when computational resources are limited.

4. Designed for Bagging

- Random forests are a **bagging** (Bootstrap Aggregation) algorithm by design. Bootstrapping ensures that each tree contributes uniquely to the ensemble, which is fundamental to bagging's success.
- Cross-validation doesn't create the same diversity among the trees because it evaluates the same dataset multiple times without the randomness of replacement.

Why do Random Forests use bootstrapping instead of cross-validation?

5. Robust to Overlap and High Variance

- Bootstrapping creates high variance among individual trees, which is desirable in random forests because the ensemble (via majority voting or averaging) reduces this variance.
- Cross-validation, on the other hand, aims to validate a single model, which is not aligned with the random forest's principle of combining multiple weak learners.

Summary -

Why Not Cross-Validation?

1. **Redundant with OOB**: Since OOB error provides a performance estimate, cross-validation is unnecessary.
2. **Higher Cost**: Cross-validation would add significant computational cost without offering much additional insight for random forests.
3. **Doesn't Enhance Diversity**: Cross-validation doesn't promote the diverse tree generation that bootstrapping does, which is central to the random forest algorithm.