**Study the solvers**

**Instructions**

In this lesson you learned about the various solvers that pair algorithms with a machine learning process to create an accurate model.

Walk through the solvers listed in the lesson and pick two.

In your own words, compare and contrast these two solvers.

What kind of problem do they address? How do they work with various data structures?

Why would you pick one over another?

**Answer**

I will look at liblinear (the default) and lbfgs

**Liblinear:**

solver='liblinear' is the default solver in scikit-learn's LogisticRegression().It is suitable for small-to-medium-sized datasets. It supports both binary and multiclass classification problems.

It implements a coordinate descent algorithm with L1 or L2 regularization and can handle sparse input data efficiently. It generally performs well when the number of samples is significantly larger than the number of features. It supports L1 penalty for feature selection and L2 penalty for regularization.

And it is well-suited for problems with high-dimensional feature spaces.

**LBFGS:**

solver='lbfgs' is an alternative solver available in LogisticRegression(). It is particularly recommended for larger datasets compared to liblinear. Just like the liblinear, it is also suitable for both binary and multiclass classification problems. It utilizes the Limited-memory Broyden-Fletcher-Goldfarb-Shanno (LBFGS) optimization algorithm. The LBFGS algorithm approximates the Hessian matrix to perform optimization. It can handle L2 regularization and performs well in terms of convergence speed and efficiency for multiclass problems.

However, it may have memory and computation limitations for extremely large datasets.

Finally, the main differences between liblinear and lbfgs lie in their optimization algorithms and performance characteristics. liblinear is suitable for small-to-medium-sized datasets, supports L1 and L2 regularization, and performs well in high-dimensional feature spaces. On the other hand, lbfgs is recommended for larger datasets, uses the LBFGS optimization algorithm, and is particularly efficient for multiclass problems. In conclusion, the size of your dataset, sparsity, and regularization requirements should all be considered when choosing between these solvers.