Project 2, PSL Fall 2024

Contributors

Olivia Dalglish Arindam Saha

We collectively worked on the model with respect to approach and implementation, as well as reviewing each other's code for logic and bugs.

Citations

We referenced the following in our implementation

- https://liangfgithub.github.io/Proj/F24_Proj2_hints_2_Python.html ("Efficient Implementation") for general approach
- https://campuswire.com/c/GB46E5679/feed/457 for SVD implementation
- https://campuswire.com/c/GB46E5679/feed/462 for quadratic Year column

Technical details

For approach, we followed the guide provided by https://campuswire.com/c/GB46E5679/feed

The script processes and predicts weekly sales data by leveraging various data preprocessing techniques coupled with an ordinary least squares linear regression model.
To start, the svd_dept function applies Singular Value Decomposition (SVD) to smooth sales data for each department. It pivots the data into a matrix with `Date` as rows and `Store` as columns, centers the matrix by subtracting column means, and performs SVD to decompose it into singular vectors. The matrix is then reduced to 8 components, reconstructed, and re-centered, resulting in a smoothed dataset. This smoothed data is returned in its original structure to preserve compatibility with downstream processes.
Next, to ensure consistency between the training and testing datasets, only shared values in specified identifier columns, `Store` and `Dept` are retained in both datasets. We find the unique pairs of (`Store` , `Dept`) and filter both train and test datasets accordingly. We then add columns `Week` , `Year` , and `Year^2` , derived from the `Date` field, where `Year` is the year, `Week` is a numerical column with range [1,52], and `Year^2` is the squared year.
Before fitting, we identify and remove redundant columns in the training data by checking for perfect linear relationships (indicated by negligible residuals in a least-squares fit). The columns identified as redundant are also removed from the test set.
trained and predicted for each (`Store` , `Dept`) combination. It merges the smoothed sales data with

the original training dataset and We iterate over all unique (`Store` , `Dept`) pairs, filter the training and tests sets on the pair values, and fit an OLS model to predict Weekly Sales. We then use the model to predict Weekly Sales for the corresponding test data.

Performance metrics

Fold	WMAE	Execution Time (s)
1	1943.344	30.970
2	1390.886	34.316
3	1392.232	34.527
4	1523.191	34.214
5	2308.423	35.113
6	1636.825	34.903
7	1615.023	35.848
8	1362.546	34.998
9	1350.826	36.705
10	1332.109	38.923
Avg.	**1585.540**	**35.052**

This was run on a Macbook Air with an M2 chip and 8GB memory.