In this Walmart stores forecasting project, 3 forecasting models are built in R.
Three models which were tried are: -

1. naïve model
2. snaive model
3. Combination of tslm model and stlf model

Out of the three models tslm produced better accuracy than snaive and naïve, with snaive as the second-best model.

The code splits the train.csv data from kaggle into train, test and 10-fold CV time series data for the models.

Best performing model is the combination of tlsm and stlf and was able to beat the 1600 threshold.

Feature engineering:

5th fold had higher error and after some investigation it became clear that a day prior to Christmas falls in the Christmas week in 2011 data whereas it falls in a week previous to Christmas in 2010. This created an issue since few days preceding to Christmas has important data and it helps if it falls in the Christmas week. I took similar approach as mentioned in the below link but I implemented my own less generic code to balance the weekly sales in the week 51 & 52.
https://github.com/davidthaler/Walmart_competition_code

I took 10% earning from 4 weeks preceding Christmas and shifted it to the next day in succession and subtract the same amount from the week in the training data.
With this approach accuracy of model 1 & 2 dipped a little bit but helped model 3.

stlf tuning:

In the third model, first 7 folds runs with tslm model and when 2 years data is available from fold 8 onwards it switches to stlf model.

Spent some time setting the s.window in stlf code and selected 7 as the parameter. The model was taking some time to converge so had to give up further tuning which might have resulted in better results.

Overall average score achieved by tslm+stlf model is **1581.917**

Running time of the 3 models combined:

user	system	Total elapsed time
1488.020	103.147	1604.516 seconds (appx 26.74 minutes)

System Used: MacBook Pro - 2.8 GHz Intel Core i7, 16 GB 2133 MHz LPDDR3

WMAE per fold and overall average:

Fold	Model 1	Model 2	Model 3
1	2078.93	2262.422	2042.456
2	2589.338	1787.081	1440.11
3	2253.936	1779.052	1434.711
4	2823.098	1716.117	1597
5	5156.012	2265.714	2197.287
6	4218.348	1696.932	1674.197
7	2269.904	2086.967	1610.574

8	2143.839	1750.283	1330.464
9	2221.145	1719.887	1270.725
10	2372.425	1680.956	1221.64
Overall Average:	**2812.697**	**1874.541**	**1581.917**