Home assignment summary:

In this assignment, I was asked to build a model to predict future sales of SKU, per store on a daily basis.

my approach:

1. I choose to predict the overall SKU sales - i.e aggregate the daily sales over all the stores. The motivation was that it should be easier to model for the overall sales, thus that is a very good starting point. Moreover, the sales of the stores are highly correlated (above 0.7) for most of the stores.
2. Modeling: I chose to use a hybrid method, with one model to predict the trend and the second model to predict the residuals, between the trend prediction and the target, which is mostly due to seasonality and the SKU properties. For the trend model, I choose linear regression and for the residuals mode, I used XGBoost.
3. **Features**:
   Trend model - I used simple time dummies for the linear regression of order 1, i.e dummy for each day of the year: trend = a*t +c
   residuals model: I used the following features: ['sku_category', 'tot_promoted', 'month', 'day_of_month', 'day_of_year', 'week_of_month', 'week_of_year', 'day_of_week', 'is_wknd', 'quarter', 'season', 'sale_amount']. Most of the above features are self-explanatory, the 'sale_amount', is a feature that ranks each SKU from 0-4, according to its overall sales, during 2016.
4. **The temporal data**: I have chosen to use training data only from 2017. Also, I have tried to predict all the SKU sales. The reason I did not focus on a subgroup of SKUs, is due to the fact that there is high variance between the different SKUs - some are rarely sold over the year, whereas others are being sold on a weekly/daily basis.
5. **Evaluation:** For the validation set, per SKU, I have calculated the following metrics:'RMSE_val', 'MAE_val', 'median_absolute_error_val', 'bias_val', 'median_sales_true_val', 'median_sales_pred_val', 'median_relative_error%_val', 'std_relative_error_val', 'sum_of_sales_val'. Most of the metrics are self-explanatory. I have added the 'sum_of_sales_val', which is the total sales of an SKU in the validation period, since i have noticed the model is having trouble accurately predicting for SKUs with low amounts of sales.

Important notes:

- Time horizon: I did not dwell on this, but if I would, I would ask what are the business objectives. There is a clear tradeoff between predicting far into the future and losing accuracy in predicting the demand. Given a constraint, relative error not above X, and a minimum time for prediction, I would plot, per product, a curve, where on the x-axis is the predicting time ahead and on the y-axis the accuracy metric, and choose accordingly.
- Pre-processing - I have chosen not to use rolling mean on the target, since, first when I tried 2- 5 days rolling mean, the validation performance decreased. Also, it seems reasonable to assume the rolling average can damage the target, for example rolling for Sunday (low volume) with prices from Saturday (high volume)
- Predicting per store - it could be that I should have started to work initially at a resolution of store-SKU. Yet, I did try to build a hierarchy model, where one model predicts per SKU and a second model, the “geo-res”, predicts the residuals between the SKU overall prediction and the SKU sales of the store. The results were poor and due to time constrain I choose to leave that approach.
- More features/models - given more time I would design more features, for example, the frequency of a given SKU sales: daily, weekly, monthly, quarterly. Clearly, I would try to optimize the models, choose other models, feature selection, etc.
- Holidays: one approach to address the holidays - a time when there is an anomaly (very high) consumption of a subset of products, was to let the model train over both 2016 and 2017, hoping that from 2016, the model will learn to identify when there are “extreme events”. However, the validation performance decreased and I left this approach.