NextHikes

Pharmaceutical Sales prediction across multiple stores

Overview

Business Need

You work at Nexthikes as a Machine Learning Engineer and a company called **Rossmann Pharmaceuticals** has given you a project on sales forecasting. The finance team wants to forecast sales in all their stores across several cities six weeks ahead of time. Managers in individual stores rely on their years of experience as well as their personal judgement to forecast sales.

The data team identified factors such as promotions, competition, school and state holidays, seasonality, and locality as necessary for predicting sales across the various stores.

Your job is to build and serve an end-to-end product that delivers this prediction to analysts in the finance team.

Data and Features

The data for this challenge can be found here.

Data fields

Most of the fields are self-explanatory. The following are descriptions for those that aren't.

Id - an Id that represents a (Store, Date) duple within the test set

Store - a unique Id for each store

Sales - the turnover for any given day (this is what you are predicting)

Customers - the number of customers on a given day

Open - an indicator for whether the store was open: 0 = closed, 1 = open

StateHoliday - indicates a state holiday. Normally all stores, with few exceptions, are closed on state holidays. Note that all schools are closed on public holidays and weekends. a = public holiday, b = Easter holiday, c = Christmas, 0 = None
SchoolHoliday - indicates if the (Store, Date) was affected by the closure of public schools
StoreType - differentiates between 4 different store models: a, b, c, d
Assortment - describes an assortment level: a = basic, b = extra, c = extended. Read more about assortment here
CompetitionDistance - distance in meters to the nearest competitor store
CompetitionOpenSince[Month/Year] - gives the approximate year and month of the time the nearest competitor was opened
Promo - indicates whether a store is running a promo on that day
Promo2 - Promo2 is a continuing and consecutive promotion for some stores: 0 = store is not participating, 1 = store is participating
Promo2Since[Year/Week] - describes the year and calendar week when the store started participating in Promo2
PromoInterval - describes the consecutive intervals Promo2 is started, naming the months the promotion is started anew. E.g. "Feb,May,Aug,Nov" means each round starts in February, May, August, November of any given year for that store

Learning Outcomes

- Technical Skills: Pandas, Matplotlib, Numpy, HTML and CSS ,Flask. Interns will also improve their code modularization skills.
- Creation of new features
- Predictive pipeline: Exploratory data analysis, data wrangling, building and fine-tuning models
- Building model using MLOps Techniques
- Deployment: Interns will know how to serve predictions in a web app.

Badges

Each week, one user will be awarded one of the badges below for the best performance in the category below.

In addition to being the badge holder for that badge, each badge winner will get +20 points to the overall score.

Visualization - quality of visualizations, understandability, skimmability, choice of visualization

Quality of code - reliability, maintainability, efficiency, commenting - in future this will be CICD/CML

Innovative approach to analysis -using latest algorithms, adding in research paper content and other innovative approaches

Writing and presentation - clarity of written outputs, clarity of slides, overall production value

Most supportive in the community - helping others, adding links, tutoring those struggling

The goal of this approach is to support and reward expertise in different parts of the Machine learning engineering toolbox.

Badges

Each week, one Intern will be awarded one of the badges below for the best performance in the category below.

In addition to being the badge holder for that badge, each badge winner will get +20 points to the overall score.

Visualization - quality of visualizations, understandability, skimmability, choice of visualization

Quality of code - reliability, maintainability, efficiency, commenting - in future this will be CICD

Innovative approach to analysis -using latest algorithms, adding in research paper content and other innovative approaches

Writing and presentation - clarity of written outputs, clarity of slides, overall production value

Most supportive in the community - helping others, adding links, tutoring those struggling

The goal of this approach is to support and reward expertise in different parts of the Data Scientist toolbox.

Late Submission Policy

Our goal is to prepare successful learners for the work and submitting late, when given enough notice, shouldn't be necessary.

For interim submissions, those submitted 1-6 hours late will receive a maximum of 50% of the total possible grade. Those submitted >6 hours late may receive feedback, but will not receive a grade.

For final submissions, those submitted 1-24 hours late, will receive a maximum of 50% of the total possible grade. Those submitted >24 hours late may receive feedback, but will not receive a grade.

Instructions

The task is divided into the following objectives

- Exploration of customer purchasing behavior
- Prediction of store sales
    - Machine learning approach
    - Deep Learning approach
- Serving predictions on a web interface

Task 1 - Exploration of customer purchasing behaviour

Exploratory data analysis is the lifeblood of every meaningful machine-learning project. It helps us unravel the nature of the data and sometimes informs how you go about modelling. A careful exploration of the data encapsulates checking all available features, checking their interactions and correlation as well as their variability with respect to the target.

In this task, you seek to explore the behaviour of customers in the various stores. Our goal is to check how some measures such as promos and opening of new stores affect purchasing behavior.

To achieve this goal, you need to first clean the data. The data cleaning process will involve building pipelines to detect and handle outlier and missing data. This is particularly important because you don't want to skew our analysis.

Visualizing various features and interactions is necessary for clearly communicating our findings. It is a powerful tool in the data science toolbox. Communicate the findings below via the necessary plots.

You can use the following questions as a guide during your analysis. It is important to come up with more questions to explore. This is part of our expectation for an excellent analysis.

- Check for distribution in both training and test sets - are the promotions distributed similarly between these two groups?
- Check & compare sales behavior before, during and after holidays
- Find out any seasonal (Christmas, Easter etc) purchase behaviours,

- What can you say about the correlation between sales and number of customers?
- How does promo affect sales? Are the promos attracting more customers? How does it affect already existing customers?
- Could the promos be deployed in more effective ways? Which stores should promos be deployed in?
- Trends of customer behavior during store open and closing times
- Which stores are opened on all weekdays? How does that affect their sales on weekends?
- Check how the assortment type affects sales
- How does the distance to the next competitor affect sales? What if the store and its competitors all happen to be in city centres, does the distance matter in that case?
- How does the opening or reopening of new competitors affect stores? Check for stores with NA as competitor distance but later on has values for competitor distance

Deliver your exploratory analysis notebook - make sure you answer all the questions asked in task 1 using the appropriate plots or summary tables and give useful insights. A 3 - 5 slides presentation is enough for interim submission.

1.2 - Logging

Log your steps using the logger library in python.

Task 2 - Prediction of store sales

Prediction of sales is the central task in this challenge. you want to predict daily sales in various stores up to 6 weeks ahead of time. This will help the company plan ahead of time.

The following steps outline the various sub tasks needed to effectively do this:

2.1 Preprocessing

It is important to process the data into a format where it can be fed to a machine learning model. This typically means converting all non-numeric columns to numeric, handling NaN values and generating new features from already existing features.

In our case, you have a few datetime columns to preprocess. you can extract the following from them:

- weekdays
- weekends
- number of days to holidays
- Number of days after holiday
- Beginning of month, mid month and ending of month
- (think of more features to extract), extra marks for it

As a final thing, you have to scale the data. This helps with predictions especially when using machine learning algorithms that use Euclidean distances. you can use the standard scaler in sklearn for this.

2.2 Building models with sklearn pipelines

At this point, all our features are numeric. Since our problem is a regression problem, you can narrow down the list of algorithms you can use for modelling.

A reasonable starting point will be to use any of the tree based algorithms. Random forests Regressor will make for a good start.

Also, for the sake of this challenge, work with sklearn pipelines. This makes modeling modular and more reproducible. Working with pipelines will also significantly reduce your workload when you are moving your setup into files for the next part of the challenge. Extra marks will be awarded for doing this.

2.3 Choose a loss function

Loss functions indicate how well our model is performing. This means that the loss functions affect the overall output of sales prediction.
Different loss functions have different use cases.

In this challenge, you're allowed to choose your own loss function. you need to defend the loss function you choose for this challenge. Feel free to be creative with your choice. You might want to use loss functions that are easily interpretable.

2.4 Post Prediction analysis

Explore the feature importance from our modelling. Creatively deduce a way to estimate the confidence interval of your predictions. Extra marks will be given for this.

2.5 Serialize models

To serve the models you built above, you need to serialize them. Save the model with the timestamp(eg. 10-08-2020-16-32-31-00.pkl). This is necessary so that you can track predictions from various models.

Assume that you'll make daily predictions. This means you'll have various models for predictions hence the reason for serializing the models in the format above.

2.6 Building model with deep learning

Deep Learning techniques can be used to predict various outcomes including but not limited to future sales. Your task is to create a deep learning model of the Long Short Term Memory which is a type of Recurrent Neural Network .

You can use either Tensorflow or Pytorch libraries for model building. The model should not be very deep (Two layers) due to the computational requirements, it should comfortably run in google colab.

1. Isolate the Rossmann Store Sales dataset into time series data
2. Check whether your time Series Data is Stationary
3. Depending on your conclusion from 2 above difference your time series data
4. Check for autocorrelation and partial autocorrelation of your data
5. Transform the time series data into supervised learning data by creating a new y(target) column. For example as illustrated here in the **Sliding Window For Time Series** Data section
6. Scale your data in the (-1, 1) range
7. Build a LSTM Regression model to predict the next sale.

2.7 Using MLFlow to serve the prediction

Use the code snippet provided by mlflow to make inference on your test data

Task 3 - Serving predictions on a web interface

Use one of the platforms of your choice (Flask, Streamlit, pure javascript, etc.) to design, and build a backend to make inference using your trained model and input parameters collected through a frontend interface.

Your dashboard should provide an easy way for a user (in this case managers of the stores) to enter required input parameters, and output the predicted sales amount and customer numbers.

The input fields in the frontend are for example

- Store_id
- Upload csv file with a columns name
  - Date
  - IsHoliday
  - IsWeekend
  - IsPromo
  - Any other parameter which is dependent on the date
- Any any other parameter requires as input for your model that is not dependent on date

Finally your dashboard should show a plot that shows the predicted sales amount and number of customers. It should also allow the user to download the prediction in the form of a csv table.

Hosting

you can use heroku for hosting or any other platform that allows you to host your App. Create a free heroku instance and deploy your code. Submit a link to your site.

Interim Submissions

To be categorised to the different weeks till 4th week

- Your employer wants a quick meeting after you've done a first quick pass of the data and wants to know whether further investigation is useful. To achieve this, summarize your findings from Exploration of customer purchasing behavior (task 1) in 3 - 5 slides.
- Link to your Github code that includes your Jupyter notebook.
- Submit Screenshots showing:
  - Multiple data versions in your DVC store
  - Multiple model versions in your MLFlow dashboard

Feedback

You may not receive detailed comments on your interim submission, but will receive a grade.

Final Submission **(3-11-2024)**

- PDF suitable to submit as a blog on your analysis. More emphasis on Deep Learning Modelling is rewarded.
- Link to your Github code that includes your Jupyter notebook.

- Link to your Deployed Application and/or screenshot of your dashboard
- Screenshots demonstrating anything else you have done

Feedback

You will receive comments/feedback in addition to a grade.