10 Academy: Artificial Intelligence Mastery

Week 4 Challenge Document

Date: 01 Jan- 14 Jan 2025

Overview

Business Need

You work at Rossmann Pharmaceuticals as a Machine Learning Engineer. 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 judgment to forecast sales.

The data team identified factors such as promotions, competition, school and state holidays, seasonality, and locality as necessary for predicting the 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. Or you can find it also here: Rossmann Store Sales | Kaggle

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 - the 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, or November of any given year for that store

Learning Outcomes

Skills:

- Advanced use of scikit-learn
- Feature Engineering
- ML Model building and fine-tuning
- CI/CD deployment of ML models
- Python logging
- Unit testing
- Building dashboards
- Model management
- MLOps with DVC, CML, and MLFlow

Knowledge:

- Reasoning with business context
- Data exploration
- Predictive analysis
- Machine learning
- Hyperparameter tuning
- Model comparison & selection

Communication:

- Reporting on statistically complex issues

Competency Mapping

The tasks you will carry out in this week's challenge will contribute differently to the 11 competencies 10 Academy identified as essential for job preparedness in the field of Data Engineering, and Machine Learning engineering. The mapping below shows the change (lift) one can obtain through delivering the highest performance in these tasks.

Competency	**Potential contributions from this week**
Professionalism for a global-level job	Articulating business values
Collaboration and Communicating	Reporting to stakeholders
Software Development Frameworks	Using Github for CI/CD, writing modular codes, and packaging
Python Programming	Advanced use of python modules such as Pandas, Matplotlib, Numpy, Scikit-learn, Prophet and other relevant python packages
SQL programming	MySQL db create, read, and write
Data & Analytics Engineering	data filtering, data transformation, and data warehouse management
MLOps & AutoML	Pipeline design, data and model versioning,
Deep Learning and Machine Learning	NLP, topic modelling, sentiment analysis
Web & Mobile app programming	HTML, CSS ,Flask, Streamlit

Team

Tutors:

- Mahlet
- Elias
- Rediet
- Kerod
- Emitinan
- Rehmet

Key Dates

- Discussion on the case - 09:00 UTC on Wednesday 01 Jan 2025. Use #all-week-4 to pre-ask questions.
- Interim Solution - 20:00 UTC on Friday 03 Jan 2025.
- Final Submission - 20:00 UTC on Tuesday 14 Jan 2025

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 behavior

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 modeling. 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 behavior of customers in the various stores. Our goal is to check how some measures such as promos and the 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 outliers 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 behaviors,
- What can you say about the correlation between sales and the 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 opening and closing times
- Which stores are open 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 centers, 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 have 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.

1.2 - Logging

Log your steps using the logger library in Python for traceability and reproducibility.

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 a holiday
- Beginning of the month, mid-month, and end of the 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 modeling.

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 modeling. 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 differences 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 an LSTM Regression model to predict the next sale.

Task 3 - Model Serving API Call

- Create a REST API to serve the trained machine-learning models for real-time predictions.
  - **Choose a framework:**
    - Select a suitable framework for building REST APIs (e.g., Flask, FastAPI, Django REST framework).
  - **Load the model:**
    - Use the serialized model from Task 2 to load the trained machine-learning model.
  - **Define API endpoints:**
    - Create API endpoints that accept input data and return predictions.
  - **Handle requests:**
    - Implement logic to receive input data, preprocess it, and make predictions using the loaded model.
  - **Return predictions:**
    - Format the predictions and return them as a response to the API call.
  - **Deployment:**
    - Deploy the API to a web server or cloud platform.

Tutorials Schedule

Overview

In the following, the colour **purple** indicates morning sessions, and **blue** indicates afternoon sessions.

Wednesday: Data Exploration

In addition to understanding the week's challenge, students will also understand how to explore time series data.

- Introduction to the challenge(Mahlet)
- Data pipeline and Time series data exploration(Elias)

Key Performance Indicators:

- Reasoning with business context
- Understanding time series data
- Relevant skill in the area - previous skill utilization
- Ability to help others

Thursday: Predictive ML

Here students will learn how to use machine learning to forecast sales.

- Model versioning, Hyperparameter tuning and cross validation(Rediet)
- Predictive Machine Learning(Emitinan)

Key Performance Indicators:

- Successful sales forecasting
- Pipeline thinking
- Efficient and modular coding technique
- Proactivity to self-learn - sharing references

Friday: Deep Learning

Here students will learn how to use Understanding time series data deep learning techniques to perform future predictions.

- Deep Learning(Kerod)

Key Performance Indicators:

- Understanding Recurrent Neural Networks and LSTM model
- Using TensorFlow or pytorch for model building
- Proactivity to self-learn - sharing references

Interim Submission

- 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 the Exploration of customer purchasing behavior (task 1). Aim for 10-20 slides.
- Link to your GitHub code that includes your Jupyter notebook.

Feedback

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

Final Submission

- PDF suitable to submit as a blog on your analysis. More emphasis on Deep Learning Modeling is rewarded.
- Link to your Github code, and make sure to screenshots demonstrating anything else you have done.

Feedback

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

References

1. Loss functions
2. Sklearn pipelines
3. Merging dataframes
4. Introduction to flask
5. HTML and CSS
6. Time series analysis
7. RandomForests

Must Read

1. Time-series forecasting of seasonal items sales using machine learning – A comparative analysis - ScienceDirect

Time Series Forecasting With LSTMs

1. Time series forecasting with LSTM
2. Univariate Time series forecasting with Deep Learning
3. Autocorrelation and Partial AutoCorrelation

MLOps

1. Auto-sklearn — AutoSklearn 0.12.7 documentation (automl.github.io)

Kaggle kernels -

1. https://www.kaggle.com/c/rossmann-store-sales/notebooks
2. https://www.kaggle.com/thie1e/exploratory-analysis-rossmann
3. https://www.kaggle.com/elenapetrova/time-series-analysis-and-forecasts-with-prophet
4. https://www.kaggle.com/shearerp/interactive-sales-visualization
5. https://www.kaggle.com/michaelpawlus/obligatory-xgboost-example
6. https://www.kaggle.com/stefanozakher94/eda-and-forecasting-with-rfregressor-final-updated
7. https://www.kaggle.com/emehdad/time-series-linear-models-tslm

8. https://www.kaggle.com/sammyshen/exploratory-and-randomforest