KDnuggets

Visit KDnuggets.com for more cheatsheets and additional learning resources.

Machine Learning with ChatGPT

With ChatGPT, building a machine learning project has never been easier. By simply writing follow-up prompts and analyzing the results, you can quickly and easily train the model to respond to user queries and provide helpful insights.

Project Planning

Keyword Methodology
>>> Write a short summary of your machine learning project idea using the following keywords: `predict customer churn`, `cosmetic data`, `logistic regression`, `accuracy and recall`. Please do not restate any part of this instruction in your summary.

>>> **Follow-up Prompt**: Please return suggested responses that are about how you could change or improve your summary

Data Description Methodology
>>> I have a cosmetic dataset consisting of 10,000 rows and 16 columns: [List of Column Names]. Can you list the steps I have to follow to develop an end-to-end customer classification project.

Feature Engineering

General Method
>>> Write a Python code to perform feature engineering

Feature Engineering with Columns
>>> Write a Python code for feature engineering using ['product_name', 'ingredients', 'price', 'allergens', 'positive_reviews', 'positive_reviews'] columns. Please add only feature engineering part.

Data Preprocessing

General
>>> write a Python code to clean and preprocess the cosmetic dataset

Data Cleaning
>>> Write a Python code to handling missing values and handling any outliers or anomalies in the data.

Preprocessing
>>> Write the Python code to convert categorical variables into numeric variables and scale/normalize the data.

Class Imbalance Issue
>>> Please address the class imbalance issue by either oversampling the minority class or undersampling the majority class.

Model Selection

General
>>> Write a Python code for model selection. Please include decision trees, random forests, logistic regression, or support vector machines (SVM).

Training and Testing sets
>>> Write a follow-up Python code to split the dataset into training and testing sets.

Evaluation Metric
>>> Try using Accuracy, F1 Score, and AUC score for model evolution in model selection section.

Hyperparameter Tuning

Model Evaluation
>>> Please write a Python code to perform model evaluation for a logistic regression model. In order to ensure that the model is not overfitting to the training data, please use cross-validation. Use accuracy, precision, recall, and F1 score as evaluation metrics.

Hyperparameter Tuning
>>> Can you modify the above code to include hyperparameter tuning and save the best-performing model?

Bayesian Optimization
>>> Can you write a Python code to tune hyperparameters using Bayesian optimization?

Experiment Tracking

Suggestion
>>> Can you suggest some popular tools for experiment tracking in machine learning projects?

Experiment Tracking with MLFlow
>>> Can you write a Python code using MLflow for experiment tracking with logistic regression? The code should include setting up the experiment, defining hyperparameters, training the model, logging metrics, and saving the trained model to MLflow. Make sure to also end the MLflow run properly after the experiment.

Tracking Code, Model, and Dataset
>>> Explain how to use DVC to track code, model, and dataset in a machine learning project. Provide a step-by-step guide, including how to add data to DVC, how to create a Python script for training a model and tracking experiments using MLflow, and how to commit changes to Git.

MLOps

Streamlit App
>>> Explain how to create a Streamlit web app for a customer classifier in Python. In addition, provide a sample code that loads a saved model and feature scaler and makes predictions based on user input.

Model Deployment on Spaces
>>> Explain how to deploy a Streamlit app to Hugging Face space without creating a Docker file. List the necessary steps, including creating a requirements.txt file and a runtime.txt file, logging in to your Hugging Face space account, and using the Hugging Face CLI to create and deploy your app.

Model Monitoring
>>> Explain how to monitor a model on a Hugging Face Streamlit app. Discuss the importance of monitoring model performance and usage, and describe the tools and techniques available for monitoring.

Automation
>>> Can you provide a detailed guide on creating a Continuous Integration and Delivery (CI/CD) pipeline using Python to regularly update the model on Hugging Face with the latest dataset? Please include steps for automating data cleaning, preprocessing, model training, and deployment.

Subscribe to KDnuggets News

Abid Ali Awan | 2023