Support Vector Machine Classification in Google Colab

Objective

In this assignment, you will practice implementing a classification problem using a Support Vector Machine (SVM) in Google Colab. You will use an advertising dataset to predict user click behavior. The main focus will be on understanding the SVM model, performing hyperparameter tuning using grid search, and evaluating the model's performance.

Instructions

Set Up Your Environment in Google Colab

- Open Google Colab and create a new notebook.

Download the Dataset

- Use the following code snippet to download the advertising dataset directly into your Colab environment:

```
url = ('https://raw.githubusercontent.com/PawelSmolinski/
       MachineLearning/main/Datasets/advertising.csv')
```

- This command downloads the dataset.

Load and Explore the Dataset

- Use pandas to load the dataset:

```
import pandas as pd
ad_data = pd.read_csv(url)
```

- Explore the dataset using methods like `ad_data.head()`, `ad_data.describe()`, etc.

Prepare the Data

- Select relevant features for the classification task.
- Split the data into training and testing sets.

Implement the SVM Model

- Define the SVM model and perform hyperparameter tuning using GridSearchCV.
- Fit the model on the training set.

Evaluate the Model

- Evaluate the model on the test set and note the performance metrics like accuracy.

Advanced Challenge (Optional)

- Implement a grid search using regularized logistic regression.
- Focus on adjusting the regularization strength (‘C‘) and the type of regularization (‘penalty‘). The ‘C‘ parameter inversely controls the strength of regularization, with smaller values specifying stronger regularization. The ‘penalty‘ parameter can typically be set to ‘l1‘ or ‘l2‘, where ‘l1‘ corresponds to Lasso regularization and ‘l2‘ corresponds to Ridge regularization.
- Since not all solvers support both ‘l1‘ and ‘l2‘ penalties, you can use the ‘liblinear‘ solver for a combination of both penalties. If you encounter issues with other solvers, it is advisable to use the ‘l2‘ penalty, which is more broadly supported.
- Sample code to set log regression right:

```
log_reg = LogisticRegression(solver='liblinear')
```

- Analyze the impact of these changes on model accuracy and complexity.
- Remember to preprocess your dataset and split it into training and testing sets before applying the grid search.

Resources

- Google Colab
- Pandas Documentation
- Scikit-learn SVM
- Scikit-learn GridSearchCV
- ChatGPT