Machine Learning

Lab 4. Overfitting and underfitting. Model evaluation metrics

In the previous exercises, we observed the functioning of several learning models, such as logistic regression. The Iris dataset used in these examples was divided into a training set and a test set. The training set was used only for training the model, not for testing it. Similarly, the test set was used only for testing, not for training. This strategy helps to properly evaluate the trained model.

Learning involves adjusting the parameters of a model to best fit the training data, but within certain reasonable limits. If the model is fitted too precisely (overfit), it will not perform well with new data. Overfitting can also occur when the training set is too small. There is also the opposite phenomenon - underfitting. In this case, the model fits too little to the structure of the data. If this is due to the properties of the model itself, we say it is biased. These two phenomena are illustrated in the program "overfitting.py." Linear regression using polynomials was implemented. The goal is to fit the model (blue line) to the data (blue dots), which are samples of a sinusoidal function, whose course is shown in orange on the displayed charts. The first-order polynomial (linear function) is a biased model - it cannot fit the data. The result is a large mean squared error (MSE) on the test data. In the case of a fourth-order polynomial, the result is better - the model fits the data well, and the MSE decreases. In the case of a fifteenth-order polynomial, which has too many degrees of freedom in this problem, the fit to the training samples is almost perfect, and overfitting occurs - the MSE is very large because the fitted function significantly differs from the actual sinusoidal function that describes the samples.

Task 1: Analyze the overfitting.py program. Identify variables corresponding to the number of training samples, the level of "noise" causing the samples to deviate from the actual distribution, and the degrees of the polynomial. What are the results of different changes to these parameters?

The number of model parameters is one of the factors that can cause overfitting, especially if there are relatively few training examples. However, there are techniques that allow to mitigate the risk of overfitting. This is where regularization comes in, which involves modifying the cost function with an appropriate term that increases the cost in case of excessive fitting of the model. Consult the literature or textbooks to learn more about regularization and cost functions (e.g., section 1.1 in Pattern Recognition and Machine Learning by Bishop).

Classification and regression quality metrics

A trained model needs to be tested (evaluated). We have different evaluation metric to use dependent of what kind of problem the model is solving (classification, regression, clustering etc.) It is worth to know a handful of them. Visit:

https://scikit-learn.org/0.15/modules/model_evaluation.html#prediction-error-metrics

to study the definitions and formulae of several metrics. Then, do the tasks below.

Task 2: write a program for classification of the Iris dataset (binary classification for two arbitrary classes, arbitrary classifier). Evaluate the classifier on the test set using the following metrics: the confusion matrix, accuracy, F1-score, Jaccarda coefficient.

Task 3. Implement linear regression for prediction of sepal width based on sepal length. Then, evaluate the regressor on the test set using the following metrics: mean squared error (MSE), mean absolute error (MAE), explained variance score, determination coefficient R^2.

This instruction was based on, amongst others:

https://scikit-learn.org/stable/auto_examples/model_selection/plot_underfitting_overfitting.html

https://scikit-learn.org/stable/modules/cross_validation.html

https://scikit-learn.org/stable/model_selection.html

https://scikit-learn.org/0.15/modules/model_evaluation.html

https://brain.fuw.edu.pl/edu/index.php/Uczenie_maszynowe_i_sztuczne_sieci_neuronowe/Wyk%C5%82ad_Ocena_jako%C5%9Bci_klasyfikacji

ChatGPT

This instruction was written in Polish by Jakub Jurek, then translated by ChatGPT.