Exercise 6 – Evaluation II

Introduction to Machine Learning

Hint: Useful libraries

R

```
# Consider the following libraries for this exercise sheet:

library(mlbench)
library(mlr3)
library(mlr3learners)
```

Python

```
# Consider the following libraries for this exercise sheet:

# general
import numpy as np
import pandas as pd

# sklearn
from sklearn.linear_model import LogisticRegression
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
from sklearn.model_selection import RepeatedKFold
from sklearn.model_selection import RepeatedStratifiedKFold
```

Exercise 1: Overfitting & underfitting

Learning goals

Discuss for given situations if they might lead to overfitting / underfitting

Assume a polynomial regression model with a continuous target variable y, a continuous, p-dimensional feature vector $\mathbf{x}$ and polynomials of degree d, i.e.,

$$f\left(\mathbf{x}^{(i)}\right) = \sum_{j=1}^{p} \sum_{k=0}^{d} \theta_{j,k} (\mathbf{x}_j^{(i)})^k.$$

For each of the following situations, indicate whether we would generally expect the performance of a flexible polynomial learner (high d) to be better or worse than an inflexible one (low d). Justify your answer.

NB

We can only state tendencies here; performance strongly depends on the specific data situation.

---

i. The sample size n is extremely large, and the number of features p is small.

---

ii. The number of features p is extremely large, and the number of observations n is small.

---

iii. The true relationship between the features and the response is highly non-linear.

---

iv. The data could only be observed with a high level of noise.

Are overfitting and underfitting properties of a learner or of a fixed model? Explain your answer.

Should we aim to completely avoid both overfitting and underfitting?

Exercise 2: Resampling strategies

Learning goals

1. Implement resampling procedures in R/Python
2. Understand how the choice of resampling strategy affects the quality of the GE estimator

---

Why would we apply resampling rather than a single holdout split?

Classify the `german_credit` data into solvent and insolvent debtors using logistic regression. Compute the training error w.r.t. MCE.

Python Hint

Read the already preprocessed file german_credit_for_py.csv

In order to evaluate your learner, compare the test MCE using

1. three times ten-fold cross validation (3x10-CV)
2. 10x3-CV
3. 3x10-CV with stratification for the feature `foreign_worker` to ensure equal representation in all folds
4. a single holdout split with 90% training data

Hint

R

You will need rsmp, resample and aggregate.

Python

You will need RepeatedKFold, RepeatedStratifiedKFold and train_test_split.

Discuss and compare your findings and compare them to the training error computed previously.

Would you consider LOO-CV to be a good alternative?