Exercise 1: Overfitting & underfitting

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

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

$\forall i \in \{1, \dots, n\}$.

a) 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.

   (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 are very noisy.

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

c) Should we aim to completely avoid both overfitting and underfitting?

Exercise 2: Resampling strategies

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

b) 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`.)

c) In order to evaluate your learner, compare test MCE using

   i) three times ten-fold cross validation (3x10-CV)

   ii) 10x3-CV

   iii) 3x10-CV with stratification for the feature `foreign_worker` to ensure equal representation in all folds

   iv) a single holdout split with 90% training data

   (`R` Hint: you will need `rsmp`, `resample` and `aggregate`.)
   (`Python` Hint: you will need `RepeatedKFold`, `RepeatedStratifiedKFold` and `train_test_split`.)

d) Discuss and compare your findings from c) and compare them to the training error from b).

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