ITAP: Homework 1

Deadline: ~~18.~~ 25. 3. 2020

ZIP the solutions and give in the file `name-surname.zip` via Učilnica. Report is not necessary, since the grade is based on the quiz.

Solution file should contain scripts where your solution is implemented. Solutions for each problem (Slo. naloga) should be put in a separate directory named `naloga<consecutive number>`. Use R since quiz will be prepared in this programming language. **When solving the problems, use the template that can be found in Učilnica next to the data and this text.** Doing so, prevents the issues with pseudo-random number generation.

If you use some code that you have found, add the reference to the source of the code. Should you have any question about the problems, feel free to ask it on the forum.

1 Nearest Neighbours

File `podatki1.csv` contains the data that consists of 8 columns:

- columns `x1`, ..., `x6` give the values of input variables x_i, $1 \leq i \leq 6$,
- column `y` gives the values of **categorical** target variable $y \in \{0, 1\}$,
- column `z` gives the predictions for y, which are interpreted as estimates of the probabilities $P(y = 1|(x_1, \ldots, x_6))$.

An example is positive if $y = 1$, and is negative otherwise.

1.1 True Positives

Count the true positive examples when the threshold value is set to $\phi = 0{,}6$.

1.2 Area Under the Curve

Compute the area under ROC (spanned by all possible threshold values).

1.3 Pomembna značilka

One of the features x_1, x_2 and x_3 separates the two classes by itself, i.e., there exist $i \in \{1, 2, 3\}$, $y_0 \in \{0, 1\}$ and $\vartheta \in \mathbb{R}$, such that $x_i \leq \vartheta \Rightarrow y = y_0$ and $x_i > \vartheta \Rightarrow y = 1 - y_0$. Actually, there exist infinitely many such threshold

values and we can chose an arbitrary one from the maximal interval $I = [\vartheta_0, \vartheta_1)$. What is the length of the interval I? (An interval is maximal if it contains every other interval of the appropriate decision thresholds.)

1.4 Modify the Data

Since the feature x_6 is categorical (with the values `a` and `b`), and the data provider found out that we cannot know its values prior to knowing the values of y, we will create a new target variable $y' \in \{0, 1\} \times \{\mathtt{a}, \mathtt{b}\}$. What is the size of the smallest of the new four classes?

1.5 Final Model

Build a model $y' = f(x_1, \ldots, x_5)$, where f is the 1-nearest neighbour method. What is the micro-recall of the model? The pseudocode for micro-recall computation is given in Algorithm 1.

Algorithm 1 Micro-recall(possible classes Y, actual values $\boldsymbol{y}$, predictions $\boldsymbol{z}$)

$n =$ length of $\boldsymbol{y}$ and $\boldsymbol{z}$
$p = 0$ # number positives
$tp = 0$ # number of true positives
za vse $a \in Y$ **do**
    $p = p \ + \ |\{i \mid \boldsymbol{y}_i = a,\ 1 \leq i \leq n\}|$
    $tp = tp \ + \ |\{i \mid \boldsymbol{y}_i = a = \boldsymbol{z}_i,\ 1 \leq i \leq n\}|$
konec zanke
return tp/p

Micro-recall for multi-class problems is therefore a generalisation of recall, for binary targets. Note that at the end, of the algorithm, we have $p = n$, therefore, some steps can be skipped.

Food for thought: a) How would you define micro-precision? b) If there is a micro version of something, its macro version should also exist. So what would be the other option for averaging recall?

2 Linear Regression

File `podatki2.csv` contains the data, given as $m + 1 = 21$ numeric columns:

- the first m columns give the values of the input variables x_i, $1 \leq i \leq m$,

- the last one gives the values of the target variable y.

2.1 Basics

Use all the data and build a predictive model by using linear regression. Compute the predictions, and let e_m be the root mean squared error (RMSE) of the model on the train data. What is the value of e_m?

2.2 Relevant Features

Let the absolute value of the coefficient for a given feature be the estimate of its importance for predicting the target variable. Let $x_{(i)}$ be the i-th most important variable (beware - in general $x_1 \neq x_{(1)}$ etc.). Let X_k be the set of the k most important features, i.e., $X_k = \{x_{(1)}, \ldots, x_{(k)}\}$, $1 \leq k \leq m$. In the case of ties (highly unlikely), include in the set X_k the one with the smaller index.

Let e_k be RMSE of the model that uses only the features X_k, for learning. Let k_0 be the smallest index, such that $e_k \leq 1{,}1e_m$. What is the value of e_{k_0}?

2.3 Additional Features

We suspect that the model can be improved by adding the squares of the features. Use linear regression to find the model

$$y = f(x_1, \ldots x_m, x_{m+1}, \ldots, x_{2m}),$$

where $x_{m+i} = x_i^2$. What is the corresponding RMSE?

2.4 Regularisation

Solve the initial problem (with the features x_i, $1 \leq i \leq m$) with regularisation, i.e., find the parameter vector $\boldsymbol{\beta}^*$, in which the minimal value of

$$\min_{\boldsymbol{\beta}} \|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}\|_2^2 + \lambda \|\boldsymbol{\beta}\|_2^2 . \tag{1}$$

is attained. Solve the upper problem by finding $\boldsymbol{X}'$ in $\boldsymbol{Y}'$, such that the value $\min_{\boldsymbol{\beta}} \|\boldsymbol{Y}' - \boldsymbol{X}'\boldsymbol{\beta}\|_2^2$ is attained in the same point as the value (1).

What is the value of $\|\boldsymbol{\beta}^*\|_2^2$, when $\lambda = 0{,}01$?

2.5 How large λ?

Let $\boldsymbol{\beta}^*(\lambda)$ be the solution of the optimisation problem (1), and let $\alpha = \|\boldsymbol{\beta}^*(0{,}01)\|_2^2$. What is the smallest λ_0, such that $\|\boldsymbol{\beta}^*(\lambda_0)\|_2^2 < \alpha/10$? (It suffices that the absolute difference between the estimate $\hat{\lambda}_0$ and the true value does not exceed 10^{-7}.)

3 Variance and More

File `podatki3.csv` contains the data which are structured the same as the Problem 2. However, in this case, we have only one descriptive variable.

3.1 Who fits best?

For every s between 1 and 10 find the optimal polynomial p_s of degree s (in terms of RMSE), which predicts y from x. What is the smallest RMSE? Let $s = s_0$ be the corresponding degree of the polynomial.

3.2 Cross Validation, Part One

Implement the function `split(data, k)`, whose imput is a data set and the number k. The function returns a partition of the data into k parts, given as sets P_j of row-indices, $1 \leq j \leq k$. If there are n rows of the data, P_j should contain all indices $i = qk + j$, for an arbitrary $q \in \mathbb{N}$ such that $1 \leq i \leq n$. For example, if $n = 8$ and $k = 3$, then $P_1 = \{1, 4, 7\}$, $P_2 = \{2, 5, 8\}$ in $P_3 = \{3, 6\}$.

If $k = 4$, what is the average value of the input variable in the subset of data that corresponds to P_1?

3.3 Cross Validation, Part Two

Let $k = 4$. For all s from the first question, repeat the following:

- Generate k pairs (U_j, T_j) (train and test sets), where T_j contains the examples with the indices from the setP_j, whereas U_j contains the others but IN THE SAME ORDER as in the initial data (for example, following the previous example and setting $j = 1$, the order of the examples in U_j should equal $2, 3, 5, 6, 8$)
- For every set U_j, find the optimal polynomial $p_{s,j}$, and compute its error $e_{s,j}$ (RMSE) on T_j. Compute $e_s = \sum_{j=1}^{k} e_{s,j}/k$.

Let the smallest error be achieved at the degree s_1. What is the value of e_{s_1}?

The order in the training data should not matter, but let's be careful ...

Variance of p_i

By using the least squares method (or, equivalently, via maximum likelihood estimation), we find the estimate $\hat{\boldsymbol{a}}$ of the parameter vector $\boldsymbol{a}$ of the actual model $y = \boldsymbol{x}^T\boldsymbol{a} + \varepsilon$, where $\varepsilon \sim N(0, \sigma^2)$. (The polynomials above belong into this context.) Under the assumption that the noise is independently realised, for every example, find the formula for variance of the predictions of the model $\hat{y}(\boldsymbol{x}_*) = \boldsymbol{x}_*^T\hat{\boldsymbol{a}}$. More precisely, calculate

$$\mathrm{Var}[\hat{y}(\boldsymbol{x}_*) \mid \mathcal{D}] \;=\; \mathbb{E}_{\boldsymbol{\varepsilon}}\left[(\hat{y}(\boldsymbol{x}_*) - \mathbb{E}_{\boldsymbol{\varepsilon}}[\hat{y}(\boldsymbol{x}_*) \mid \mathcal{D}])^2 \mid \mathcal{D}\right],$$

where $\mathcal{D} = (\boldsymbol{X}, \boldsymbol{Y})$ denotes the observed data, i.e., the actual matrix $\boldsymbol{X}$ of the values of the input variables and the vector $\boldsymbol{Y}$ of the values of the target variable.

Hint: express $\hat{\boldsymbol{a}}$ with $\boldsymbol{X}$ and $\boldsymbol{Y}$. Take into account that $\hat{\boldsymbol{a}}$ is unbiased estimate of $\boldsymbol{a}$. Maybe, the relation $b^2 = bb^T$ (for scalars b) turns out to be useful.

Food for thought: a) Variance is independent of the target values. Interesting, isn't it? b) Did we need the normality assumption for the noise?

3.4 Variance p_{s_0}

Let's assume $\sigma = 1.0$. What is the variance of the prediction $p_{s_0}(1)$ (p_{s_0} is the polynomial from 3.1)?

3.5 Varianca p_{s_1}

Let's assume $\sigma = 1.0$. What is the variance of the prediction $p_{s_1}(1)$ (p_{s_1} is the polynomial from 3.3)?