Exercise Collection – ML Basics

Contents

Lecture exercises

Exercise 1: ML tasks

Identify which type of machine learning (supervised or unsupervised? What type of task?) could be used in these cases:

a) When crossing the alps using the Brenner Autobahn, there is the option to pay electronically in advance. When approaching the toll station, the barrier automatically opens when the number plate was recognised. The recognition happens automatically by a digital camera system.

b) Diagnose whether a patient suffers from cancer or not.

c) The owner of an internet site wants to protect his system against various violations of the terms of service (bot programs, manipulation of timestamps, etc.)

d) An online shopping portal wants to determine products that are automatically offered to registered customers upon login.

e) We want to sort our news into different groups.

f) We want to sort our Email into Spam/Non-Spam.

g) In a supermarket, products that are often bought together are said to be placed side by side on a shelf to increase the sales.

h) We want to extract a list of skills from XING.

i) We want to know our top customers (i. e. highest sales, logistics, etc.).

Solution 1:

a) multiclass classification (plate digits) (supervised learning)

b) binary classification (supervised)

c) outlier detection ((un)supervised)

d) frequent pattern mining (unsupervised)

e) classification (supervised) / clustering (unsupervised)

f) classification (supervised)

g) clustering / assocation rules (unsupervised)

h) not a machine learning task

i) not a machine learning task

Exercise 2: simple regression problem

Suppose we observe 6 data pairs and want to describe the underlying relationship between target y and feature $\mathbf{x}$.

x	0.56	0.22	1.7	0.63	0.36	1.2
y	160	150	175	185	165	170

a) Assume a standard linear relationship

$$y^{(i)} = \beta_0 + \beta_1 \mathbf{x}^{(i)} + \epsilon^{(i)}$$

with iid errors $\epsilon^{(i)}$ and calculate the least squares estimator $\hat{\boldsymbol{\beta}}$ for $\boldsymbol{\beta} = (\beta_0, \beta_1)^\top$ manually (+ calculator).

b) Assume a non-linear relationship (polynomial degree 2)

$$y^{(i)} = \beta_0 + \beta_1 \mathbf{x}^{(i)} + \beta_2 (\mathbf{x}^{(i)})^2 + \epsilon^{(i)}$$

with iid errors $\epsilon^{(i)}$ and calculate the least squares estimator $\hat{\boldsymbol{\beta}}$ for $\boldsymbol{\beta} = (\beta_0, \beta_1, \beta_2)^\top$ with R.

Solution 2:

a) We use the least squares-estimator introduced in the lecture: $\hat{\beta} = (X^T X)^{-1} X^T y$ with

$$X = \begin{bmatrix} 1 & x_{1,1} & x_{1,2} & ... & x_{1,m} \\ 1 & x_{2,1} & x_{2,2} & ... & x_{2,m} \\ \vdots & \vdots & \vdots & ... & \vdots \\ 1 & x_{n,1} & x_{n,2} & ... & x_{n,m} \end{bmatrix}$$

$$x = \begin{bmatrix} 0.56 \\ 0.22 \\ 1.7 \\ 0.63 \\ 0.36 \\ 1.2 \end{bmatrix}, X = \begin{bmatrix} 1 & 0.56 \\ 1 & 0.22 \\ 1 & 1.7 \\ 1 & 0.63 \\ 1 & 0.36 \\ 1 & 1.2 \end{bmatrix} \text{ and } y = \begin{bmatrix} 160 \\ 150 \\ 175 \\ 185 \\ 165 \\ 170 \end{bmatrix}$$

Then

$$\hat{\beta} = (X^T X)^{-1} X^T y$$

$$= \left(\begin{bmatrix} 1 & 1 & 1 & ... & 1 \\ x_{1,1} & x_{2,1} & x_{3,1} & ... & x_{n,1} \\ \vdots & \vdots & \vdots & ... & \vdots \\ x_{1,m} & x_{2,m} & x_{3,m} & ... & x_{n,m} \end{bmatrix} \begin{bmatrix} 1 & x_{1,1} & x_{1,2} & ... & x_{1,m} \\ 1 & x_{2,1} & x_{2,2} & ... & x_{2,m} \\ \vdots & \vdots & \vdots & ... & \vdots \\ 1 & x_{n,1} & x_{n,2} & ... & x_{n,m} \end{bmatrix} \right)^{-1} \begin{bmatrix} 1 & 1 & 1 & ...1 & \\ x_{1,1} & x_{2,1} & x_{3,1} & ... & x_{n,1} \\ \vdots & \vdots & \vdots & ... & \vdots \\ x_{1,m} & x_{2,m} & x_{3,m} & ... & x_{n,m} \end{bmatrix} \begin{bmatrix} 160 \\ 150 \\ 175 \\ 185 \\ 165 \\ 170 \end{bmatrix}$$

$$= \left(\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 0.56 & 0.22 & 1.7 & 0.63 & 0.36 & 1.2 \end{bmatrix} \begin{bmatrix} 1 & 0.56 \\ 1 & 0.22 \\ 1 & 1.7 \\ 1 & 0.63 \\ 1 & 0.36 \\ 1 & 1.2 \end{bmatrix} \right)^{-1} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 0.56 & 0.22 & 1.7 & 0.63 & 0.36 & 1.2 \end{bmatrix} \begin{bmatrix} 160 \\ 150 \\ 175 \\ 185 \\ 165 \\ 170 \end{bmatrix}$$

$$= \begin{bmatrix} 6 & 4.67 \\ 4.67 & 5.2185 \end{bmatrix}^{-1} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 0.56 & 0.22 & 1.7 & 0.63 & 0.36 & 1.2 \end{bmatrix} \begin{bmatrix} 160 \\ 150 \\ 175 \\ 185 \\ 165 \\ 170 \end{bmatrix}$$

$$= \begin{bmatrix} 0.5491944 & -0.4914703 \\ -0.4914703 & 0.6314394 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 0.56 & 0.22 & 1.7 & 0.63 & 0.36 & 1.2 \end{bmatrix} \begin{bmatrix} 160 \\ 150 \\ 175 \\ 185 \\ 165 \\ 170 \end{bmatrix}$$

$$= \begin{bmatrix} 0.2739710 & 0.4410709 & -0.2863051 & 0.23956809 & 0.3722651 & -0.04056998 \\ -0.1378643 & -0.3525536 & 0.5819766 & -0.09366351 & -0.2641521 & 0.26625693 \end{bmatrix} \begin{bmatrix} 160 \\ 150 \\ 175 \\ 185 \\ 165 \\ 170 \end{bmatrix}$$

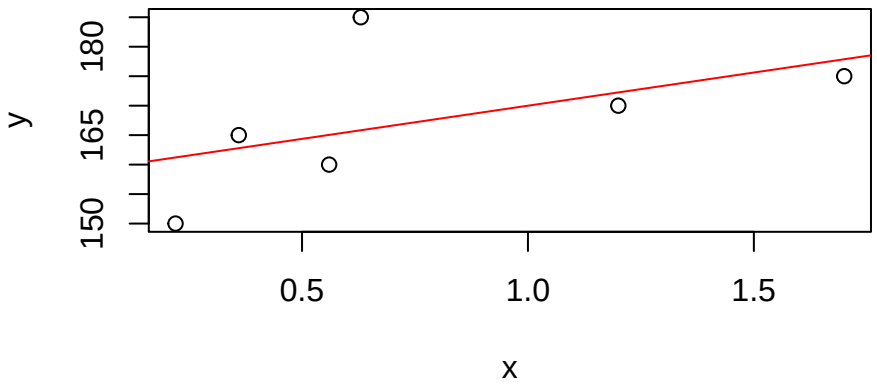
$$= \begin{bmatrix} 158.73954 \\ 11.25541 \end{bmatrix}$$

Hence the linear model $\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x = 158.73954 + 11.25541x$

```
x = c(0.56, 0.22, 1.7, 0.63, 0.36,1.2)
y = c(160,150,175,185,165,170)

X <- sapply(0:1, function(k) x^k)
solve(t(X) %*% X) %*% t(X) %*% y
```

```
##            [,1]
## [1,] 158.73954
## [2,]  11.25541
```

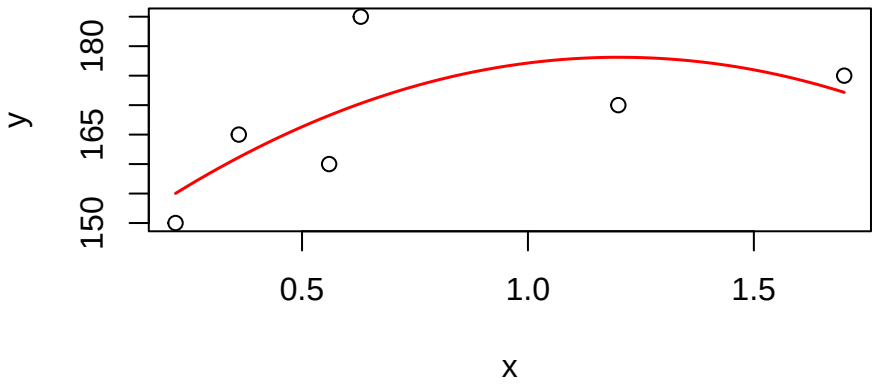


b) Here $X = \begin{bmatrix} 1 & 0.56 & 0.3136 \\ 1 & 0.22 & 0.0484 \\ 1 & 1.7 & 2.89 \\ 1 & 0.63 & 0.3969 \\ 1 & 0.36 & 0.1296 \\ 1 & 1.2 & 1.44 \end{bmatrix}$ and $\hat{\beta} = \begin{bmatrix} 143.51682 \\ 57.59155 \\ -23.96347 \end{bmatrix}$

```
x = c(0.56, 0.22, 1.7, 0.63, 0.36,1.2)
y = c(160,150,175,185,165,170)

X <- sapply(0:2, function(k) x^k)
solve(t(X) %*% X) %*% t(X) %*% y

##             [,1]
## [1,] 143.51681
## [2,]  57.59155
## [3,] -23.96347
```



Exercise 3: credit scoring project

Imagine you work at a bank and have the job to develop a credit scoring model. This means, your model should predict whether a customer applying for a credit will be able to pay it back in the end.

a) Is this a supervised or unsupervised learning problem? Justify your answer.

b) How would you set up your data? Which is the target variable, what feature variables could you think of? Do you need labeled or unlabeled data? Justify all answers.

c) Is this a regression or classification task? Justify your answer.

d) Is this "learning to predict" or "learning to explain"? Justify your answer.

e) In classical statistics, you could use e.g. the logit model for this task. This means we assume that the targets are conditionally independent given the features, so $y^{(i)}|\mathbf{x}^{(i)} \perp y^{(j)}|\mathbf{x}^{(j)}$ for all $i, j = 1, \ldots, n, i \neq j$, where n is the sample size. We further assume that $y^{(i)}|\mathbf{x}^{(i)} \sim Bin(\pi^{(i)})$, where $\pi^{(i)} = \frac{\exp(\boldsymbol{\theta}^\top \mathbf{x}^{(i)})}{1+\exp(\boldsymbol{\theta}^\top \mathbf{x}^{(i)})}$. Looking at this from a Machine Learning perspective, write down the hypothesis space for this model. State explicitly which parameters have to be learned.

f) In classical statistics, you would estimate the parameters via Maximum Likelihood estimation. (The log-Likelihood of the Logit-Model is: $\sum_{i=1}^{n} y^{(i)} \log(\pi^{(i)}) + (1 - y^{(i)})(\log(1 - \pi^{(i)}))$). How could you use the model assumptions to define a reasonable loss function? Write it down explicitly.

g) Now you have to optimize this risk function to find the best parameters and hence the best model. Describe with a few sentences, how you could do this.

Congratulations, you just designed your first Machine Learning project!

Solution 3:

a) Supervised learning problem - the model will be learned from historical credit data for which payment history has been observed (knowing the ground truth is vital here since we need to evaluate our model's accuracy)

b) Target variable: classes (default y/n), continuous credit scores, or class probabilities). Potential features: monthly income, current level of indebtedness, past credit behavior, profession, residential environment, age, number of kids etc. Labels: yes, since we have a supervised learning problem.

c) This is a classification problem - we want to assign our customers to classes *default* and *non-default*.

d) (Primarily) learning to predict - we want to score future borrowers.

e) $\mathcal{H} = \{\pi : \mathcal{X} \mapsto [0, 1] \mid \pi(\mathbf{x} \mid \boldsymbol{\theta}) = s(\boldsymbol{\theta}^T \mathbf{x}), \boldsymbol{\theta} \in \mathbb{R}^d\}$, where $s(z) = 1/(1 + exp(-z))$ is the sigmoid function. Parameters to be learned: $\boldsymbol{\theta}$.

f) We know that, in the optimum, (log-)likelihood is maximal. We can directly translate this into risk minimization by using the *negative* log-likelihood as our empirical risk. We will just use the pointwise negative log-likelihood as our loss function:

$$L\left(y^{(i)}, \pi\left(\mathbf{x}^{(i)} \mid \boldsymbol{\theta}\right)\right) = -\left(y^{(i)} \log\left(\pi\left(\mathbf{x}^{(i)} \mid \boldsymbol{\theta}\right)\right) + \left(1 - y^{(i)}\right)\left(\log\left(1 - \pi\left(\mathbf{x}^{(i)} \mid \boldsymbol{\theta}\right)\right)\right)\right)$$

(the so-called *Bernoulli loss*). The empirical risk is then the sum of point-wise losses:

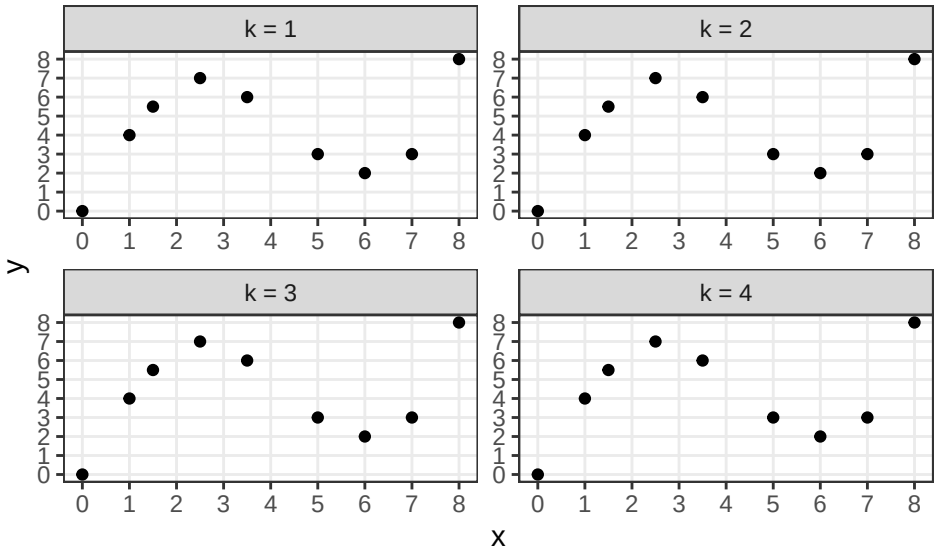
$$\mathcal{R}_{\text{emp}}(\boldsymbol{\theta}) = -\sum_{i=1}^{n} y^{(i)} \log\left(\pi\left(\mathbf{x}^{(i)} \mid \boldsymbol{\theta}\right)\right) + \left(1 - y^{(i)}\right)\left(\log\left(1 - \pi\left(\mathbf{x}^{(i)} \mid \boldsymbol{\theta}\right)\right)\right)$$

g) We can now solve this optimization problem via empirical risk minimization, which, in this case, is perfectly equivalent to ML estimation. Therefore, we set the first derivative of $\mathcal{R}_{\text{emp}}(\boldsymbol{\theta})$ wrt $\boldsymbol{\theta}$ to 0 and solve for $\boldsymbol{\theta}$. However – unlike linear regression – this has no closed-form solution, so a numerical optimization procedure such as gradient descent is required.

Further exercises

Exercise 4: WS2020/21, second, question 1

ID	x	y
1	0.0	0.0
2	1.0	4.0
3	1.5	5.5
4	2.5	7.0
5	3.5	6.0
6	5.0	3.0
7	6.0	2.0
8	7.0	3.0
9	8.0	8.0



Now we want to train a cubic polynomial, i.e., a polynomial regression model with degree $d = 3$ on the data used in a).

(i) Define the hypothesis space of this model and state explicitly how many parameters have to be estimated for training the model.

(ii) Define the minimization problem that we have to optimize in order to train the polynomial regression model. Use L2 loss and be as explicit as possible - without plugging in the data.

(iii) In order to estimate the parameters of the model, it is convenient to describe the model as a linear model. Compute the respective design matrix using the concrete values of $\mathbf{x}$ given above. Additionally, state a formula for estimating the parameters using this design matrix. (You do not have to derive this formula.)

Solution 4:

(i)

$$\mathcal{H} = \{f : f(\mathbf{x}) = \theta_0 + \theta_1 \mathbf{x} + \theta_2 \mathbf{x}^2 + \theta_3 \mathbf{x}^3 \mid \theta_0, \theta_1, \theta_2, \theta_3 \in \mathbb{R}\}$$

The four parameters $\theta_0, \theta_1, \theta_2, \theta_3$ have to be estimated

(ii)

$$\hat{\boldsymbol{\theta}} \in \arg\min_{\boldsymbol{\theta} \in \Theta} \mathcal{R}_{\text{emp}}(\boldsymbol{\theta})$$

This means we have to optimize the following minimization problem wrt $\boldsymbol{\theta} = (\theta_0, \theta_1, \theta_2, \theta_3)$:

$$\min_{\boldsymbol{\theta} \in \Theta} \mathcal{R}_{\text{emp}}(\boldsymbol{\theta}) = \min_{\boldsymbol{\theta} \in \Theta} \sum_{i=1}^{9} (y^{(i)} - (\theta_0 + \theta_1 \mathbf{x}^{(i)} + \theta_2 (\mathbf{x}^{(i)})^2 + \theta_3 (\mathbf{x}^{(i)})^3))^2$$

(iii)

$$\hat{\boldsymbol{\theta}} = (X^\top X)^{-1} X^\top y$$

X1	x	x.2	x.3
1	0.0	0.00	0.000
1	1.0	1.00	1.000
1	1.5	2.25	3.375
1	2.5	6.25	15.625
1	3.5	12.25	42.875
1	5.0	25.00	125.000
1	6.0	36.00	216.000
1	7.0	49.00	343.000
1	8.0	64.00	512.000

Exercise 5: WS2020/21, second, question 6

Describe a real-life application in which classification might be useful and where we want to "learn to explain". Describe the response, as well as the predictors. Explain your answer thoroughly.

Solution 5:

No model solution

Ideas & exercises from other sources