MAD 2020-21, Assignment 2

Bulat Ibragimov, Kim Steenstrup Pedersen

hand in until: 1.12.2020 at 23:59

General comments: The assignments in MAD must be completed and written individually. You are allowed (and encouraged) to discuss the exercises in small groups. If you do so, you are required to list your group partners in the submission. The report must be written completely by yourself. In order to pass the assignment, you will need to get at least 40% of the available points. The data needed for the assignment can be found in the assignment folder that you download from Absalon.

Submission instructions: Submit your report as a PDF, not zipped up with the rest. Please add your source code to the submission, both as executable files and as part of your report. To include it in your report, you can use the `lstlisting` environment in LaTeX, or you can include a "print to pdf" output in your pdf report.

Exercise 1 (**Weighted Average Loss, 4 points, based on Exercise 1.11 in Rogers & Girolami).** The following expression is known as the **weighted** average loss:

$$\mathcal{L} = \frac{1}{N}\sum_{n=1}^{N} \alpha_n (f(\mathbf{x}_n; \mathbf{w}) - t_n)^2 = \frac{1}{N}\sum_{n=1}^{N} \alpha_n \left(\mathbf{w}^T\mathbf{x}_n - t_n\right)^2$$

where the influence of each data point is controlled by its associated weight $\alpha_n > 0$ parameter.

a) (2 points): Assuming that each α_n is given a fixed value, show by mathematical derivation that the optimal least squares parameter value is $\hat{\mathbf{w}} = (\mathbf{X}^T\mathbf{A}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{A}\mathbf{t}$, where $\mathbf{A}$ is a diagonal matrix that contains the weights $\alpha_1, \ldots, \alpha_N$ on the diagonal. Hint: Start by rephrasing the loss function in matrix-vector form (as done in Rogers & Girolami Sec. 1.3) using the weight matrix $\mathbf{A}$, then compute the gradient of the loss function with respect to the parameters $\mathbf{w}$.

b) (2 points): Similar to Exercises 3 and 4 of Assignment 1, implement a corresponding regression model in Python (e.g., implement a new `linweighreg.py` module by making a copy of `linreg.py` and adding the code you find relevant). Afterwards, fit a model on all the features in the training set `boston_train.csv` using $\alpha_n = t_n^2$. Compute corresponding predictions for the test instances given in `boston_test.csv` and generate a scatter plot as in Exercises 3 and 4 of Assignment 1. What do you expect to happen? What do you observe? Do the additional weights have an influence on the outcome?

Deliverables. a) Derivation for the optimal solution (similarly to the lecture). Note that you do not have to show that the computed solution is a global minimum; just providing the solution and the corresponding derivation is enough. (b) The scatter plot as well as short answers to the questions raised (2-3 lines each). Add your source code to your submission.

Solution:

1. The main issue is to write up the loss function using the A matrix. The objective can be written as

$$\mathcal{L}(\mathbf{w}) = \frac{1}{N}\sum_{n=1}^{N} \alpha_n (f(\mathbf{x}_n; \mathbf{w}) - t_n)^2 = \frac{1}{N}(\mathbf{X}\mathbf{w} - \mathbf{t})^T\mathbf{A}(\mathbf{X}\mathbf{w} - \mathbf{t}),$$

where $\mathbf{X}$ and $\mathbf{t}$ contain the training points as rows and labels, respectively. The diagonal matrix $\mathbf{A} = \text{diag}(\alpha_1, \ldots, \alpha_N) \in \mathbb{R}^{N\times N}$ contains the weights on the diagonal. One can simplify the objective in the following way:

$$\mathcal{L}(\mathbf{w}) = \frac{1}{N}\mathbf{w}^T\mathbf{X}^T\mathbf{A}\mathbf{X}\mathbf{w} - \frac{2}{N}\mathbf{w}^T\mathbf{X}^T\mathbf{A}\mathbf{t} + \frac{1}{N}\mathbf{t}^T\mathbf{A}\mathbf{t}$$

The gradient is then given by

$$\nabla\mathcal{L}(\mathbf{w}) = \frac{2}{N}\mathbf{X}^T\mathbf{A}\mathbf{X}\mathbf{w} - \frac{2}{N}\mathbf{X}^T\mathbf{A}\mathbf{t}$$

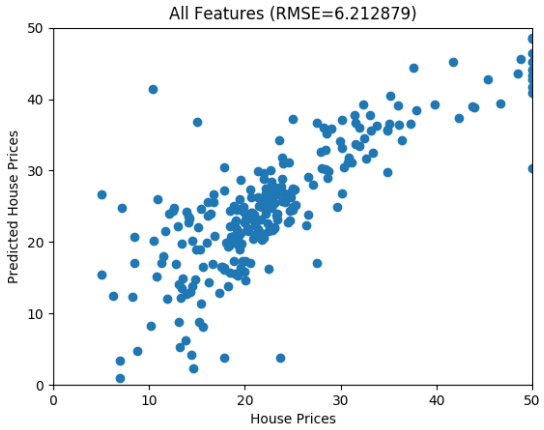

Figure 1: Scatter Plot (Weighted Linear Regression)

Hence, we have

$$\begin{aligned} & \tfrac{2}{N}\mathbf{X}^T\mathbf{A}\mathbf{X}\mathbf{w} - \tfrac{2}{N}\mathbf{X}^T\mathbf{A}\mathbf{t} &&= \mathbf{0} \\ \Leftrightarrow \quad & \mathbf{X}^T\mathbf{A}\mathbf{X}\mathbf{w} &&= \mathbf{X}^T\mathbf{A}\mathbf{t} \end{aligned}$$

An optimal solution is therefore given by $\hat{\mathbf{w}} = (\mathbf{X}^T\mathbf{A}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{A}\mathbf{t}$. Instead of isolating for $\hat{\mathbf{w}}$ it is also ok, if they substitute the given expression in th the gradient expression and show that this is zero.

2. See `linweighreg.py` and `exercise_1.py` for the code. Intuitively, using weights $\alpha_n = t_n^2$ should lead to the model focusing more on the training instances having a large value t_n. The scatter plot (for the test patterns) also indicates that this happens during the training phase. In particular, the model seems to yield a better fit for the test instances having a true value of 50 (rightmost points) and a worse fit for the instances with smaller t_n, see Figure 1.

Instructions:
2 points for (a), 2 points for (b). For (a), only deriving the solution $\hat{\mathbf{w}}$ is required (not the proof for optimality).

Exercise 2 (Polynomial Fitting with Regularized Linear Regression and Cross-Validation, 4 points). In this exercise, you will apply leave-one-out cross-validation to study the influence of the regularization parameter λ on the predictive performance of regularized linear regression. In the `men-olympic-100.txt` file, you will find data on the men's Olympic 100m running times (Hint: You can read the file into a numpy array using this numpy function, `raw = np.genfromtxt('men-olympics-100.txt', delimiter=' ')`). Take the first place running times (second column of the `raw` table) as target values $t_1, \ldots, t_{27}$. The input variables $x_1, \ldots, x_{27}$ are given in the first column (of the `raw` table).

a) (3 points): Apply polynomial fitting with regularized linear regression to fit a **first order polynomial** to the data. Plot the leave-one-out cross- validation error (y-axis) as a function of λ for some values $\lambda \in [0, 1]$ (x-axis); use `numpy.logspace(-8, 0, 100, base=10)` to generate the λ values to be tested. Report the best value of λ and the regression coefficients $\mathbf{w}$ corresponding to both $\lambda = 0$ (no regularization) and the best value of λ.

Hint: Be aware of the fact that the values outputted via `print` *are usually rounded. Make use of a higher precision when printing, e.g., via* `print("lam=%.10f and loss=%.10f" % (lam, loss))`.

b) (1 point): Repeat the same for fitting a **fourth order polynomial** to the data.

Deliverables. a) The plot, best value of λ, and the two sets of regression coefficients; b) the plot, best value of λ, and the two sets of regression coefficients. Add your source code to your submission.

Solution:

a) Fitting a regularized first order polynomial to the running times. The LOOCV errors as function of λ are shown in Figure 2. The optimal value of λ is $\lambda^* = 0.0000003430$. At $\lambda = 0$, the weights are $\mathbf{w} = (36.4164559, -0.0133308857)^T$. At λ^*, the weights are $\mathbf{w} = (36.3776282, -0.0133110046)^T$.

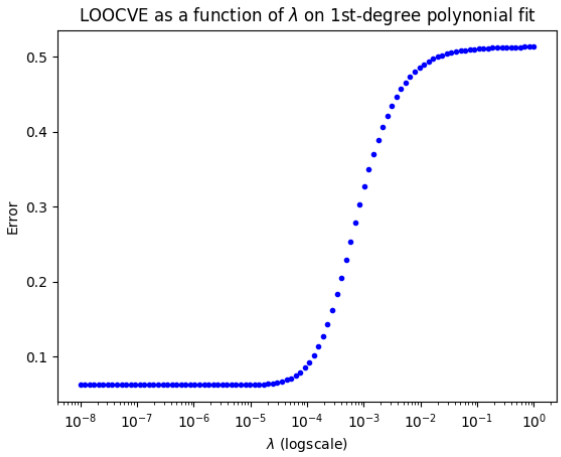

Figure 2: LOOCV error for regularized 1st order polynomial as function of λ.

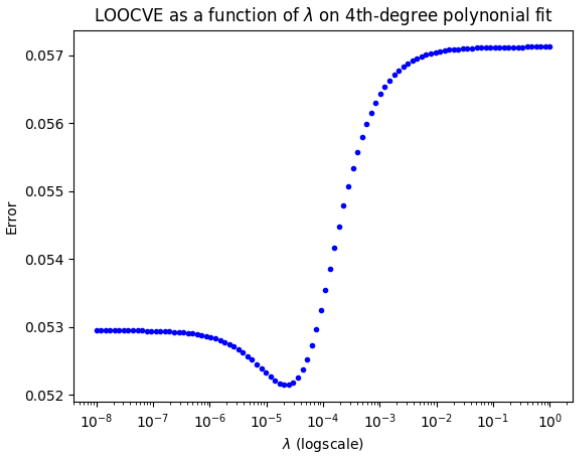

Figure 3: LOOCV error for regularized 4th order polynomial as function of λ.

b) The LOOCV errors as function of λ for a fourth order polynomial are shown in Figure 3. The optimal value is $\lambda^* = 0.0000205651$.

- At $\lambda = 0$, the weights are:

  $\mathbf{w} = (3.21358282 \times 10^5, -6.46374641 \times 10^2, 4.87449415 \times 10^{-1}, -1.63339180 \times 10^{-4}, 2.05197882 \times 10^{-8})^T$

- At λ^*, the weights are:

  $\mathbf{w} = (1.88300350 \times 10^{-2}, 9.11277821, -1.38600662 \times 10^{-2}, 7.03376807 \times 10^{-6}, -1.19035006 \times 10^{-9})^T$

Note: The values might differ depending on how the coefficients are computed (e.g., via `numpy.linalg.solve` *or via* `numpy.linalg.inv`*) and on the order of the computations conducted (e.g., order of multiplications). This is not to be considered an error in the solution.*

Instructions:
(a) 3 points, (b) 1 point

Exercise 3 (Pdf and cdf, 4 points). We model the life span x of a chip (in years) with a distribution that has the following cumulative distribution function (cdf).

$$F(x) = \begin{cases} 0 & x \leq 0 \\ 1 - exp(-\beta x^{\alpha}) & x > 0 \end{cases}$$

with $\alpha > 0$ and $\beta > 0$ being parameters.

a) (1 point): Determine the probability density function (pdf) of the distribution.
b) (2 points): Suppose we fix the parameters to $\alpha = 2$ and $\beta = \frac{1}{4}$. What is the probability that the chip works longer than four years? What is the probability that the chip stops working in the time interval [5; 10] years?
c) (1 point): How large is the median of a life span (for general choices of α, β)?

Deliverables. a) Besides the pdf also Include the derivation steps needed to go from the cdf to the pdf, b) besides the results also include the steps showing how you compute the probabilities, c) include both result and derivation steps.
Instructions:
a) 1 point b) 2 points, 1 point for each sub-question c) 1 points
Solution:

a) We need to differentiate $F(x)$. $\frac{dF}{dx} = \beta\alpha x^{\alpha-1} exp(-\beta x^{\alpha})$, $x > 0$.
b) Q1: $P(x > 4) = 1 - F(x = 4) = 1 - (1 - exp(-\beta x^{\alpha})) = exp(-\frac{1}{4}4^2) = 0.018$.
Q2: $P(x \in [5; 10] = F(x = 10) - F(x = 5) = 1 - exp(-\frac{1}{4}10^2) - (1 - exp(-\frac{1}{4}5^2)) = exp(-\frac{1}{4}5^2) - exp(-\frac{1}{4}10^2)$ $= 0.0019$
c) The median is given by the 50% percentile, we therefore equate the CDF with 0.5 and isolate for x, $1 - exp(-\beta x^{\alpha}) = 0.5 \Rightarrow exp(-\beta x^{\alpha}) = 0.5 \Rightarrow x = \frac{ln(0.5)}{-\beta}^{\frac{1}{\alpha}}$.

Exercise 4 (Conditional probability and expectations, 4 points). Professor James Duane from Regent University, USA advocates for constant execution of the right to remain silent and immediately calling the lawyer every time a person is questioned by the police on any topic. He has a popular YouTube lecture "Don't talk to the Police" and the book "You Have the Right to Remain Innocent". Let's try to probabilistically evaluate his claims using the following model.

We have individuals with no history of convictions (person NC) and with a history of convictions (person C). Suppose:

1) If any person (either NC or C) remains silent, there is a 0.001 risk of his case ending up in the court.
2) If a person NC talks to the police, there is a 0.002 risk of the police using his words against him and his case ending up in the court.
3) If a person C talks to the police, there is a 0.005 risk of the police using his words against him and his case ending up in the court.
4) The probability of not being convicted in the court are 0.5 for person NC and 0.1 for person C.
5) The probability of not being convicted drops 4 times if a person did not talk to the police during the investigation, as the jury thinks he has something to hide.
6) If a person is convicted and talked to the police during the investigation, his sentence is reduced by a 0.75 factor as he is considered cooperative.

The police arrested a person suspecting he is involved in a certain crime. The penalty for the crime is 5 years in prison:

a) (2 points): A person has no history of convictions (NC). What are the expected mean sentence durations he will have to spend in prison if convicted in the two cases he talks to the police or if he remains silent?
b) (2 points): A person has a history of convictions (C). What are the expected mean sentence durations he will have to spend in prison if convicted in the two cases he talks to the police or if he remains silent?

Deliverables. For both questions include your answer and derivation steps.
Instructions:
a) 2 point b) 2 points
Solution:
We know:

1) $P(\text{court}|\neg\text{talked}) = 0.001$
2) $P(\text{court}|\text{talked}, \text{NC}) = 0.002$
3) $P(\text{court}|\text{talked}, \text{C}) = 0.005$
4) $P(\neg\text{convicted}|\text{court}, \text{NC}) = 0.5$
$P(\neg\text{convicted}|\text{court}, \text{C}) = 0.1$

5) $P(\neg\text{convicted}|\text{court}, \neg\text{talked}, \text{NC}) = \frac{1}{4}P(\neg\text{convicted}|\text{court}, \text{NC}) = \frac{1}{8}$
$P(\neg\text{convicted}|\text{court}, \neg\text{talked}, \text{C}) = \frac{1}{4}P(\neg\text{convicted}|\text{court}, \text{C}) = \frac{1}{40}$
6) If convicted and talked, sentence is reduced by a 0.75 factor.

The penalty for the crime is 5 years, then:

a) The probability of being convicted for person NC who remains silent:
$P(\text{convicted}|\text{court}, \neg\text{talked}, \text{NC}) = 1 - P(\neg\text{convicted}|\text{court}, \neg\text{talked}, \text{NC}) = \frac{7}{8}$
$P(\text{convicted} \wedge \text{court}|\neg\text{talked}, \text{NC}) = P(\text{convicted}|\text{court}, \neg\text{talked}, \text{NC})P(\text{court}|\neg\text{talked}) = \frac{7}{8} \cdot 0.001 = \frac{7}{8000} = 0.000875$.
The probability of being convicted for person NC who talks to the police
$P(\text{convicted}|\text{court}, \text{NC}) = 1 - P(\neg\text{convicted}|\text{court}, \text{NC}) = 0.5$
$P(\text{convicted} \wedge \text{court}|\text{talked}, \text{NC}) = P(\text{convicted}|\text{court}, \text{NC})P(\text{court}|\text{talked}, \text{NC}) = 0.5 \cdot 0.002 = \frac{1}{1000} = 0.001$.
The mean sentence is therefore $= 0.000875 * 5 + 0.001 * 0.75 * 5 = 0.008125$ year, or 3 days.

b) By analogy, for person C:
The probability of being convicted for person C who remains silent:
$P(\text{convicted}|\text{court}, \neg\text{talked}, \text{C}) = 1 - P(\neg\text{convicted}|\text{court}, \neg\text{talked}, \text{C}) = \frac{39}{40}$
$P(\text{convicted} \wedge \text{court}|\neg\text{talked}, \text{C}) = P(\text{convicted}|\text{court}, \neg\text{talked}, \text{C})P(\text{court}|\neg\text{talked}) = \frac{39}{40} \cdot 0.001 = \frac{39}{40000} = 0.000975$.
The probability of being convicted for person C who talks to the police
$P(\text{convicted}|\text{court}, \text{C}) = 1 - P(\neg\text{convicted}|\text{court}, \text{C}) = 0.9$
$P(\text{convicted} \wedge \text{court}|\text{talked}, \text{C}) = P(\text{convicted}|\text{court}, \text{C})P(\text{court}|\text{talked}, \text{C}) = 0.9 \cdot 0.005 = 0.0045$.
The mean sentence is therefore $= 0.000975 * 5 + 0.0045 * 0.75 * 5 = 0.02175$ years, or 8 days.