

HOMEWORK SET 0

CSCI5527 Deep Learning (Spring 2026)

Due 11:59 pm, Feb 07 2026

Instruction Your writeup, either typeset or scanned, should be a single PDF file. For problems requiring coding, organize all codes for each top-level problem (i.e., Problem 1, Problem 2, etc) into a separate Jupyter notebook file (i.e., .ipynb file). Your submission to Gradescope should include the single PDF and all notebook files—**please DO NOT zip them!** No late submission will be accepted. For each problem, you should acknowledge your collaborators—**including AI tools**, if any.

About the use of AI tools You are strongly encouraged to use AI tools—they are becoming our workspace friends. It takes a bit of practice to ask the right and effective questions/prompts to these tools; we highly recommend that you go through this popular free short course **ChatGPT Prompt Engineering for Developers** offered by <https://learn.deeplearning.ai/> to get started.

If you use any AI tools for any of the problems, you should include screenshots of your prompting questions and their answers in your writeup. The answers provided by such AI tools often contain factual errors and reasoning gaps. **So, if you only submit an AI answer with such bugs for any problem, you will obtain a zero score for that problem.** You obtain the scores only when you explain the bugs and also correct them in your own writing. You can also choose not to use any of these AI tools, in which case we will grade based on the efforts you have made.

Notation Please refer to our [supplementary notes on high-dimensional calculus](#)

Important notes You should carefully study our [supplementary notes on high-dimensional calculus](#) before trying to answer any questions here. If you struggle too much with this homework set, it is strongly suggested that you either spend intensive efforts during the first few weeks to build up the background or consider taking a later iteration of the course when you feel ready.

Please provide detailed steps with justification for all problems; jumping to the final results leads to a zero score. Also, if we ask you to use certain facts/tools to obtain something, you have to use these facts/tools (perhaps plus others); otherwise, you get a zero score, e.g., if you prove everything from scratch when we ask you to use an existing theorem.

Problem 1 (Matrix norms, inner products, traces; 7/15) Recall that for any vector $v \in \mathbb{R}^n$, the ℓ_p norm of v is defined as $\|v\|_p \doteq (\sum_i |v_i|^p)^{1/p}$. The cases where $p = 1, 2, \infty$ are often used. When $p = 2$, it is also called the Euclidean norm. Similar norms can be defined for matrices. Particularly, the direct generalization of the vector Euclidean norm is the *Frobenius norm* defined as

$$\|M\|_F \doteq \sqrt{\sum_{ij} m_{ij}^2}$$

for a matrix M . On the other hand, the inner (i.e., dot) product of matrices is defined similarly to that of vectors. For A, B of the same size, $\langle A, B \rangle \doteq \sum_{ij} a_{ij} b_{ij}$. Obviously, $\langle A, B \rangle = \langle B, A \rangle$ and $\|M\|_F = \sqrt{\langle M, M \rangle}$. A third notion of interest is the matrix trace, $\text{tr}(M) = \sum_i m_{ii}$, i.e., sum of the diagonal entries, which is only defined for square matrices.

- (a) Show that $\langle A, B \rangle = \text{tr}(A^\top B) = \text{tr}(B^\top A)$, and then conclude that $\|M\|_F = \sqrt{\text{tr}(M^\top M)}$.
(1/15)

- (b) Assume A and B have the same size. In general, $A^\top B$ and BA^\top have different sizes, but $\text{tr}(A^\top B) = \text{tr}(BA^\top)$. Use the fact in (a) to show it! (1/15)
- (c) Use the fact in (b) to show that $\text{tr}(M_1 M_2 M_3) = \text{tr}(M_3 M_1 M_2) = \text{tr}(M_2 M_3 M_1)$, assuming that the sizes of M_1 , M_2 and M_3 are compatible with all the matrix multiplications. This is known as the *cyclic property* of the matrix trace. (1/15)
- (d) For any matrices A, B, C, D of compatible sizes, we always have $\langle ACB, D \rangle = \langle CB, A^\top D \rangle = \langle AC, DB^\top \rangle$, i.e., we can always move the **leading** matrix of one side of the inner product to the other side as **leading** matrix **once transposed**, and similarly the **trailing** matrix to the other side as **trailing** matrix **once transposed**. Use the cyclic property of the matrix trace to show this. (1/15)
- (e) Let Q be an orthonormal matrix (i.e., all columns of Q have unit norm and are mutually orthogonal to each other, $Q^\top Q = I$). Show that $\|QM\|_F = \|M\|_F$. (1/15)
- (f) Let A, B, C be three matrices of the same size, and \odot denote the Hadamard (i.e., elementwise) product ([https://en.wikipedia.org/wiki/Hadamard_product_\(matrices\)](https://en.wikipedia.org/wiki/Hadamard_product_(matrices))) between two matrices. Show that $\langle A \odot B, C \rangle = \langle A, B \odot C \rangle = \langle B, A \odot C \rangle$. (1/15)
- (g) A square matrix P is an orthogonal projector if and only if $P^2 = P$ and $P^\top = P$. Consider $A \in \mathbb{R}^{m \times n}$ with full column rank. Show that $H \doteq A(A^\top A)^{-1}A^\top$ is an orthogonal projector. (1/15)

Problem 2 (Gradient and Hessian, chain rules, perturbation-expansion method; 6.5/15)

- (a) Consider a binary classification problem with matrix input and $\{+1, -1\}$ binary labels, for which a training set $\{(\mathbf{X}_i, y_i)\}_{i=1}^N$ is given. Consider the following empirical risk minimization formulation formed using the log loss

$$f(\mathbf{B}) = \frac{1}{N} \sum_{i=1}^N \log(1 + e^{-y_i \langle \mathbf{B}, \mathbf{X}_i \rangle}).$$

Derive $\nabla f(\mathbf{B})$ using the **perturbation-expansion method**. (Hint: by the single-variable Taylor's theorem, $\log(1 + e^{z+\delta}) = \log(1 + e^z) + \frac{e^z}{1+e^z} \delta + O(\delta^2)$ as $\delta \rightarrow 0$). Validate your results either by analytic derivation using the chain rule or by any AI tools or by <https://www.matrixcalculus.org/>; show your derivation or screenshots. (1.5/15)

- (b) In (a), $f(\mathbf{B})$ takes a matrix as input and so its Hessian $\nabla^2 f(\mathbf{B})$ is most naturally represented as a 4-th order tensor (i.e., 4-dimensional array): this is because each element in \mathbf{B} has two indices, and so any of the second-order partial derivatives takes the form of

$$\frac{\partial^2 f}{\partial b_{ij} \partial b_{k\ell}},$$

which may be daunting to work with in practice. Fortunately, for the purpose of numerical computation, it is often sufficient to work with $\nabla^2 f(\mathbf{B})[\mathbf{V}]$: here, \mathbf{V} is a given matrix (direction) and the output $\nabla^2 f(\mathbf{B})[\mathbf{V}]$ is another matrix—both matrices have the same size as that of \mathbf{B} , and we should think of $\nabla^2 f(\mathbf{B})[\mathbf{V}]$ as the “Hessian tensor” $\nabla^2 f(\mathbf{B})$ takes in \mathbf{V} and returns a matrix. To derive $\nabla^2 f(\mathbf{B})[\mathbf{V}]$, we can apply our usual perturbation-expansion method to $f(\mathbf{B} + \delta \mathbf{V})$ and then compare it to

$$f(\mathbf{B} + \delta \mathbf{V}) = f(\mathbf{B}) + \delta \langle \nabla f(\mathbf{B}), \mathbf{V} \rangle + \frac{1}{2} \delta^2 \langle \mathbf{V}, \nabla^2 f(\mathbf{B})[\mathbf{V}] \rangle + o(\delta^2)$$

to extract out the corresponding $\nabla^2 f(\mathbf{B})[\mathbf{V}]$ term. **Follow this idea** to derive $\nabla^2 f(\mathbf{B})[\mathbf{V}]$ for $f(\mathbf{B})$ in (a). (Hint: by the single-variable Taylor's theorem, $\log(1 + e^{z+\delta}) = \log(1 + e^z) + \frac{e^z}{1+e^z}\delta + \frac{1}{2}\frac{e^z}{(1+e^z)^2}\delta^2 + O(\delta^3)$ as $\delta \rightarrow 0$). Validate your results either by analytic derivation using the chain rule (warning: this is going to be a prolonged path) or by any AI tools or by <https://www.matrixcalculus.org/>; show your derivation or screenshots. (1.5/15)

- (c) Consider the regression problem based on a one-layer network: $g(\mathbf{W}) = \|\mathbf{y} - \sigma(\mathbf{W}\mathbf{x})\|_2^2$, where the activation $\sigma = \tanh$ is applied elementwise to the vector $\mathbf{W}\mathbf{x}$. Derive $\nabla g(\mathbf{W})$ using whatever methods you are comfortable with. (Hint: If following the perturbation-expansion idea, you may want to first show that $\sigma((\mathbf{W} + \Delta)\mathbf{x}) = \sigma(\mathbf{W}\mathbf{x}) + \sigma'(\mathbf{W}\mathbf{x}) \odot (\Delta\mathbf{x}) + o(\|\Delta\|_F)$ when $\Delta \rightarrow \mathbf{0}$, and then plug it back into the square and expand the square using the identity $\|\mathbf{a} + \mathbf{b} + \mathbf{c}\|_2^2 = \|\mathbf{a}\|_2^2 + \|\mathbf{b}\|_2^2 + \|\mathbf{c}\|_2^2 + 2\langle \mathbf{a}, \mathbf{b} \rangle + 2\langle \mathbf{a}, \mathbf{c} \rangle + 2\langle \mathbf{b}, \mathbf{c} \rangle$. Keep in mind that to derive the gradient, any terms of order lower than $\|\Delta\|_F$ are not interesting.) Validate your results either by analytic derivation using the chain rule (warning: this is going to be a prolonged path) or by any AI tools or by <https://www.matrixcalculus.org/>; show your derivation or screenshots. (1.5/15)
- (d) Consider $f(\mathbf{x}) = \frac{1}{2}\|\mathbf{y} - \mathbf{A}\mathbf{x}\|_2^2 + \lambda\|\mathbf{x}\|_2^2$ with $\lambda > 0$. Any \mathbf{x}_0 is a candidate minimizer of $f(\mathbf{x})$ if $\nabla f(\mathbf{x}_0) = \mathbf{0}$ (i.e., the first-order optimality condition). Please find all candidate minimizers of $f(\mathbf{x})$. How many can you find in total? Please include your detailed calculation and reasoning; you can use whatever way you want to derive $\nabla f(\mathbf{x})$ if necessary. (Hint: recall that a symmetric matrix \mathbf{M} is positive definite if and only if $\mathbf{v}^\top \mathbf{M} \mathbf{v} > 0$ for all $\mathbf{v} \neq \mathbf{0}$; all positive definite matrices are invertible) (1.5/15)
- (e) Continuing with (d). What happens when $\lambda < 0$? (0.5/15)

Problem 3 (Conditional probability and Bayes' Rule; 1.5/15) Let $P(A | B)$ be the probability of event A given event B . Bayes' theorem states that

$$P(A | B) = \frac{P(B | A) P(A)}{P(B)}.$$

In general, for a partition $\{A_i\}$ of the sample space,

$$P(A_i | B) = \frac{P(B | A_i) P(A_i)}{\sum_i P(B | A_i) P(A_i)}.$$

- (a) Suppose that women who often drink alcohol are 10 times more likely to develop stomach cancer than women who do not, while the corresponding risk for men is 20 (drinker v.s. non-drinker). Statistical data show that 40% of women drink alcohol often. Now, a woman has been diagnosed with stomach cancer. But due to some privacy issues, it is impossible for you to get other information about this woman. Based on the limited information, is it possible to calculate the probability that she is a drinker? If yes, what is the probability? If not, explain why and what additional information you will need.¹ (1/15)
- (b) Doctors apply a standardized test for a certain disease. If the patient has the disease, the test shows a positive result with a probability of 99%. However, with 2% probability, a healthy patient can have a positive test result. Statistical data show that 1 of 1000 in the population have the disease. What is the probability that a patient with a positive test is affected by the disease? (0.5/15)

¹All data and information in this question are made up. No medical studies/institutions have proven it.