

ASSIGNMENT 5

P1. (15 points) The temperature at any point in the plane is given by the function

$$T(x, y) = \frac{100}{x^2 + y^2 + 1}.$$

- (a) What shape are the level curves of T ?
- (b) Where on the plane is it hottest? What is the temperature at that point?
- (c) Find the direction of the greatest increase in temperature at the point $(3, 2)$. What is the magnitude of that greatest increase?
- (d) Find the direction of the greatest decrease in temperature at the point $(3, 2)$.
- (e) Find a direction at the point $(3, 2)$ in which the temperature does not increase or decrease.

P2 (20 points). Consider the following function:

$$f(x, y) = 10x^2y - 5x^2 - 4y^2 - x^4 - 2y^4.$$

- (a) Find and classify the critical points of the function (into local minimum, local maximum, or saddle points). Also, find the corresponding function values at the critical points.
- (b) Plot the level curves of $f(x, y)$ using matplotlib's `contour()` function. Please make sure you have level curves passing nearby each of the five critical points. You can do this by setting the parameter `levels` for the `contour()` function to be an array of values, e.g., `levels=[-40, -20, -1, 3]`. Your chosen array for `levels` should contain values near the function values at the critical points.

Hint: You will need the optimality conditions.

Necessary Conditions: Let $w^* \in \mathbb{R}^n$ be an unconstrained local minimum of $f : \mathbb{R}^n \rightarrow \mathbb{R}$. Assume f is continuously differentiable in an open set S containing w^* . Then, $\nabla f(w^*) = 0$.

If in addition f is twice continuously differentiable within S , then $\nabla^2 f(w^*)$ is positive semidefinite.

Sufficient Conditions: Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$ be twice continuously differentiable in an open set S . Suppose there is a vector $w^* \in S$ satisfying the conditions

$$\nabla f(w^*) = 0, \quad \nabla^2 f(w^*) \text{ is positive definite.}$$

Then, w^* is a strict unconstrained local minimum of f .

You also need the following facts to test whether a symmetric matrix is positive definite, negative definite, positive semidefinite or negative semidefinite.

Sylvester's Criterion: (i) A symmetric matrix is positive definite if and only if all its **leading** principal minors are positive. (ii) A symmetric matrix is positive-semidefinite if and only if all its principal minors are nonnegative.

The k th **leading principal minor** of a square matrix M (not necessarily symmetric) is the determinant of its upper-left $k \times k$ sub-matrix.

Example: For a 2×2 matrix $M = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$, there are two leading principal minors, which are: a and $\det(M) = ad - bc$.

A **principal minor** of a square matrix M is the determinant of a sub-matrix of M obtained by striking out the rows and columns of M of the same set of indices.

Example: For a 2×2 matrix $M = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$, there are three principal minors, which are: a , $\det(M) = ad - bc$, and d .

At some point, you will need to find the roots of a third degree polynomial. You can use `numpy.roots()` for the numerical computation.

P3. (10 points) For an $n \times n$ matrix $A = (a_{ij})$, let M_{ij} the determinant of the $(n-1) \times (n-1)$ matrix after removing the i th row and j th column of A . M_{ij} is called a **minor**. Then, for every i ,

$$\det(A) = \sum_{j=1}^n (-1)^{i+j} a_{ij} M_{ij}. \quad (1)$$

The above expression of a matrix determinant in terms of the determinants of smaller matrices is called **Laplace expansion**.

Consider the following A matrix:

$$A = \begin{pmatrix} 2 & 1 & 2 & 6 \\ 6 & 4 & 3 & 5 \\ 8 & 1 & 4 & 9 \\ 9 & 8 & 7 & 8 \end{pmatrix}.$$

Use Python (`numpy.linalg.det()`) to verify the formula for Laplace expansion by expanding along the first row, i.e., $i = 1$.

P4. (10 points) Consider the following A matrix:

$$A = \begin{pmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{pmatrix}.$$

Write down and expand the characteristic equation of the matrix A , which should give a third-degree polynomial. Find its eigenvalues by finding the roots of the characteristic equation.

From the eigenvalues, can you tell if the matrix A is positive definite? What is the condition number of A ?

P5. (10 points) Consider the following A matrix:

$$A = \begin{pmatrix} 6 & 5 & 3 \\ 4 & 1 & 5 \\ 6 & 3 & 6 \\ 5 & 3 & 6 \\ 6 & 6 & 3 \end{pmatrix}.$$

Use the linear algebra tools in `numpy` to answer the following.

- What are the singular values of A ?
- What are the eigenvalues of $A^T A$? How are these eigenvalues related to the singular values of A ?
- Based on the answer of (a), what is the rank of A ?
- If, for a given vector b , the equation $Ax = b$ has a solution, is the solution unique?
- Let $b = (-1, 2, 0, 3, 1)^T$. For the equation $Ax = b$, is there a solution to the equation? If yes, is it unique? If not, we can find an x that minimizes $\|Ax - b\|$. What is the value of minimizing x ?

P6. (5 points) Suppose A and B are $n \times n$ matrices and $x \in \mathbb{R}^n$ are the variables. What is the gradient of the dot product of Ax with Bx ? What about the Hessian?

P7. (30 points) Implement the mini-batch gradient descent algorithm for linear regression. In your code, please keep track of the training results at each step of the algorithm iteration, including the estimated parameters θ and the training error (MSE) under the current θ .

Apply your code to the given data. The input is in the file named `X.csv`; the output is in the file named `y.csv`.

Load the data from the files. Run the mini-batch gradient descent algorithm for batch sizes equal to 1, 5 and 10, separately.

At the end of the algorithm iterations, show the final fitted results for the three batch sizes: 1, 5 and 10.

Also, plot the traces of the estimated model parameters. Please do so in a 2-d plot, that is, plot the sequence of points of (θ_0, θ_1) that you collected, where θ_0 is the intercept and θ_1 is the slope. Please do so for the three different batch sizes: 1, 5, and 10.

Also, plot the training errors as a function of the iteration index for the three batch sizes.

From the above two types of plots, you should be able to see the effects of increasing the batch size.

Hint: You can build on the code that we discussed during the lectures, which can be downloaded from the github page: <https://github.com/ageron/handson-ml2>.