1 Multivariate Gaussians: A review

Multivariate Gaussian distributions crop up everywhere in machine learning, from priors on model parameters to assumptions on noise distributions. Being able to manipulate multivariate Gaussians also becomes important for analyzing correlations in data and preprocessing it for better regression and classification. We want to make sure to first cover the MVG fundamentals here.

Note that the probability density function of a non-degenerate (i.e. the covariance matrix is positive definite and, thus, invertible) multivariate Gaussian RV with mean vector, $\boldsymbol{\mu} \in \mathbb{R}^2$, and covariance matrix, $\Sigma \in \mathbb{R}^{2\times 2}$, is:

$$f(\mathbf{z}) = \frac{1}{\sqrt{(2\pi)^2 |\Sigma|}} \exp\left(-\frac{1}{2}(\mathbf{z}-\boldsymbol{\mu})^\top \Sigma^{-1}(\mathbf{z}-\boldsymbol{\mu})\right)$$

(a) Consider a two dimensional, zero mean random variable $Z = \begin{bmatrix} Z_1 & Z_2 \end{bmatrix}^\top \in \mathbb{R}^2$. In order for the random variable to be jointly Gaussian, a necessary and sufficient condition which we call the *first characterization* is that

- Z_1 and Z_2 are each marginally Gaussian, and
- $Z_1|Z_2 = z$ is Gaussian, and $Z_2|Z_1 = z$ is Gaussian.

A *second characterization* of a jointly Gaussian zero mean RV $Z \in \mathbb{R}^2$ is that it can be written as $Z = AX$, where $X \in \mathbb{R}^2$ is a collection of i.i.d. standard normal RVs and $A \in \mathbb{R}^{2\times 2}$ is a matrix.

Let X_1 and X_2 be i.i.d. standard normal RVs. Let U denote a binary random variable uniformly that is equal to 1 with probability $\frac{1}{2}$ and -1 with probability $\frac{1}{2}$, independent of everything else.

For each of the below subproblems, complete the following *two* steps: (1) Using one of the characterizations given above, determine whether the RVs are jointly Gaussian. If using the second characterization, clearly specify the A matrix. (2) Calculate the covariance matrix of Z (regardless of whether the RVs are jointly Gaussian or not).

(i.) $Z_1 = X_1$ and $Z_2 = X_2$.

(ii.) $Z_1 = X_1$ and $Z_2 = X_1 + 2X_2$. If using the first characterization, assume that you already know $(Z_1|Z_2 = z)$ is Gaussian.

(iii.) $Z_1 = X_1$ and $Z_2 = -X_1$.

(iv.) $Z_1 = X_1$ and $Z_2 = UX_1$.

Solution:
(i.) Based on the first characterization, Z_1 and Z_2 are jointly Gaussian because both the marginal and conditional distributions are Gaussian. $Z_1 = X_1$ and $Z_2 = X_2$ are marginally Gaussian because X_1 and X_2 are independent standard normal random variables. The conditional distributions $Z_1 \mid Z_2$ and $Z_2 \mid Z_1$ are also Gaussian, since Z_1 and Z_2 are independent.

Now using the second characterization: $Z = AX$ gives $\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = A \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ so that $A = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$. Note that we also have $\Sigma_X = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$.

Finally, the covariance matrix of Z is calculated as: $\Sigma_Z = A\Sigma_X A^T$. Thus $\Sigma_Z = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

(ii.) Based on the first characterization, Z_1 and Z_2 are jointly Gaussian because both the marginal and conditional distributions are Gaussian. $Z_1 = X_1$ is marginally Gaussian because X_1 is a standard normal random variable. $Z_2 = X_1 + 2X_2$ is also marginally Gaussian because it is a linear combination of two independent Gaussian random variables X_1 and X_2, both of which are independent standard normal random variables. The conditional distribution $Z_1 \mid Z_2$ is Gaussian, as stated in the problem, which satisfies the first characterization.

Now using the second characterization: $Z = AX$ where $Z = \begin{bmatrix} Z_1 \\ Z_2 \end{bmatrix} = \begin{bmatrix} X_1 \\ X_1 + 2X_2 \end{bmatrix}, \quad X = \begin{bmatrix} X_1 \\ X_2 \end{bmatrix}$. This implies the matrix A is: $A = \begin{bmatrix} 1 & 0 \\ 1 & 2 \end{bmatrix}$. Note that we also have the covariance matrix of X: $\Sigma_X = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$.

Finally, the covariance matrix of Z is calculated as: $\Sigma_Z = A\Sigma_X A^T = \begin{bmatrix} 1 & 0 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 0 & 2 \end{bmatrix}$. Multiplying the matrices: $\Sigma_Z = \begin{bmatrix} 1 & 0 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 0 & 2 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 5 \end{bmatrix}$. Thus, the covariance matrix is: $\Sigma_Z = \begin{bmatrix} 1 & 1 \\ 1 & 5 \end{bmatrix}$.

(iii.) Based on the first characterization, Z_1 and Z_2 are jointly Gaussian because both the marginal and conditional distributions are Gaussian. $Z_1 = X_1$ is marginally Gaussian because X_1 is a standard normal random variable. $Z_2 = -X_1$ is also marginally Gaussian because it is a linear transformation of the Gaussian random variable X_1. The conditional distribution $Z_1 \mid Z_2$ is Gaussian, as Z_1 and Z_2 are linearly dependent, which satisfies the first characterization.

Now using the second characterization: $Z = AX$ where $Z = \begin{bmatrix} Z_1 \\ Z_2 \end{bmatrix} = \begin{bmatrix} X_1 \\ -X_1 \end{bmatrix}, \quad X = \begin{bmatrix} X_1 \\ X_2 \end{bmatrix}$. This implies the matrix A is: $A = \begin{bmatrix} 1 & 0 \\ -1 & 0 \end{bmatrix}$. Note that we also have the covariance matrix of X: $\Sigma_X = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$.

Finally, the covariance matrix of Z is calculated as: $\Sigma_Z = A\Sigma_X A^T = \begin{bmatrix} 1 & 0 \\ -1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & -1 \\ 0 & 0 \end{bmatrix}$. Multiplying the matrices: $\Sigma_Z = \begin{bmatrix} 1 & 0 \\ -1 & 0 \end{bmatrix} \begin{bmatrix} 1 & -1 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$. Thus, the covariance matrix is: $\Sigma_Z = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$.

(iv.) Based on the first characterization, are not jointly Gaussian because $Z_2 = UX_1$ involves the binary random variable U, which is not Gaussian. Even though $Z_1 = X_1$ is marginally Gaussian, Z_2, which is UX_1, is not purely Gaussian because U is discrete (taking values ±1 with equal probability). The conditional distributions $Z_1 \mid Z_2$ and $Z_2 \mid Z_1$ depend on the value of U, so the first characterization is not fully satisfied, and the random variables are not jointly Gaussian.

Now using the second characterization: $Z = AX$ where $Z = \begin{bmatrix} Z_1 \\ Z_2 \end{bmatrix} = \begin{bmatrix} X_1 \\ UX_1 \end{bmatrix}, \quad X = \begin{bmatrix} X_1 \\ X_2 \end{bmatrix}$. This cannot be expressed as a linear transformation of the vector $X = \begin{bmatrix} X_1 \\ X_2 \end{bmatrix}^T$ with a fixed matrix A, because U is a random variable and not a constant. Therefore, the second characterization does not apply directly in this case.

Covariance Matrix Calculation: Even though the variables are not jointly Gaussian, we can still compute the covariance matrix of Z. Let's compute the covariance matrix of $Z = \begin{bmatrix} Z_1 \\ Z_2 \end{bmatrix} = \begin{bmatrix} X_1 \\ UX_1 \end{bmatrix}$. The covariance matrix Σ_Z is given by: $\Sigma_Z = \text{Cov}(Z_1, Z_2) = \mathbb{E}\left[ZZ^T\right] - \mathbb{E}\left[Z\right]\mathbb{E}\left[Z^T\right]$. We first compute the individual components of the covariance matrix. The variance of Z_1 (which is X_1) is $\text{Var}(Z_1) = \text{Var}(X_1) = 1$. The variance of Z_2 (which is UX_1) is $\text{Var}(Z_2) = \text{Var}(UX_1) = \text{Var}(X_1) = 1$, since U takes values ± 1 with equal probability and is independent of X_1. The covariance between Z_1 and Z_2 is $\text{Cov}(Z_1, Z_2) = \mathbb{E}[Z_1 Z_2] = \mathbb{E}[X_1 \cdot UX_1] = \mathbb{E}[U]\mathbb{E}[X_1^2]$. Since $\mathbb{E}[U] = 0$ (because U takes values ± 1 with equal probability), we have $\text{Cov}(Z_1, Z_2) = 0$. Thus, the covariance matrix is $\Sigma_Z = \begin{bmatrix} \text{Var}(Z_1) & \text{Cov}(Z_1, Z_2) \\ \text{Cov}(Z_1, Z_2) & \text{Var}(Z_2) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$.

(b) Show that two Gaussian random variables can be uncorrelated, but not independent (*Hint: use one of the examples in part (a)*). On the other hand, show that two uncorrelated, jointly Gaussian RVs are independent.

Solution:
To show that two Gaussian random variables can be uncorrelated but not independent, we can use Example 1(a)(iv). Firstly, $\text{Cov}(Z_1, Z_2) = 0$ shows that Z_1 and Z_2 are uncorrelated. Secondly, knowing Z_2 gives information about U, and knowing U provides information about Z_1, violating independence.

To show that two uncorrelated, jointly Gaussian random variables are independent, we can prove via direct proof. Let X and Y be two jointly Gaussian random variables. The joint distribution of X and Y is given by the bivariate normal distribution: $f_{X,Y}(x, y) = \frac{1}{2\pi\sigma_X\sigma_Y\sqrt{1-\rho^2}} \exp\left(-\frac{1}{2(1-\rho^2)}\left(\frac{x^2}{\sigma_X^2} + \frac{y^2}{\sigma_Y^2} - 2\rho\frac{xy}{\sigma_X\sigma_Y}\right)\right)$, where σ_X^2 and σ_Y^2 are the variances of X and Y, and ρ is the correlation coefficient between X and Y. If X and Y are uncorrelated, then $\rho = 0$, and the joint distribution simplifies to: $f_{X,Y}(x, y) = \frac{1}{2\pi\sigma_X\sigma_Y} \exp\left(-\frac{1}{2}\left(\frac{x^2}{\sigma_X^2} + \frac{y^2}{\sigma_Y^2}\right)\right)$. This can be factored into the product of the marginal distributions of X and Y: $f_{X,Y}(x, y) = \frac{1}{\sqrt{2\pi\sigma_X^2}} \exp\left(-\frac{x^2}{2\sigma_X^2}\right) \cdot \frac{1}{\sqrt{2\pi\sigma_Y^2}} \exp\left(-\frac{y^2}{2\sigma_Y^2}\right) = f_X(x)f_Y(y)$. Thus, X and Y are independent. Conclusion: If two jointly Gaussian random variables are uncorrelated, then they are independent.

(c) With the setup in (a), let $Z = VX$, where $V \in \mathbb{R}^{2\times 2}$, and $Z, X \in \mathbb{R}^2$. What is the covariance matrix Σ_Z? If X is not a multivariate Gaussian but has the identity matrix $I \in \mathbb{R}^{2\times 2}$ as its covariance matrix, is your computed Σ_Z still the covariance of Z?

Solution:
For a linear transformation $Z = VX$, the covariance matrix of Z is given by: $\Sigma_Z = V\Sigma_X V^T$. We are given that the covariance matrix of X is the identity matrix $\Sigma_X = I$. Substituting this into the formula: $\Sigma_Z = VIV^T = VV^T$.

For non-Gaussian random variables, the covariance matrix Σ_X (which is given as the identity matrix) still describes how the components of X are related. Since the linear transformation $Z = VX$ only involves multiplying the vector X by a constant matrix V, the covariance structure of Z is still described by $\Sigma_Z = VV^T$, even if X is not Gaussian. The key fact here is that the covariance matrix describes the second-order moments (variances and covariances) of random variables, regardless of their distribution. Thus, the fact that X is not Gaussian does not affect the computation of Σ_Z, so $\Sigma_Z = VV^T$ is still the correct covariance matrix of Z.

(d) Given a jointly Gaussian zero mean RV $Z = [Z_1\ Z_2]^\top \in \mathbb{R}^2$ with covariance matrix $\Sigma_Z = \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{12} & \Sigma_{22} \end{bmatrix}$, derive the conditional distribution of $(Z_1|Z_2 = z)$.

Hint: The following identity may be useful

$$\begin{bmatrix} a & b \\ b & c \end{bmatrix}^{-1} = \begin{bmatrix} 1 & 0 \\ -\frac{b}{c} & 1 \end{bmatrix} \begin{bmatrix} \left(a - \frac{b^2}{c}\right)^{-1} & 0 \\ 0 & \frac{1}{c} \end{bmatrix} \begin{bmatrix} 1 & -\frac{b}{c} \\ 0 & 1 \end{bmatrix}$$

Solution:

Firstly, applying the matrix inversion formula:

$$\Sigma_Z^{-1} = \begin{bmatrix} 1 & 0 \\ -\frac{\Sigma_{12}}{\Sigma_{22}} & 1 \end{bmatrix} \begin{bmatrix} \left(\Sigma_{11} - \frac{\Sigma_{12}^2}{\Sigma_{22}}\right)^{-1} & 0 \\ 0 & \frac{1}{\Sigma_{22}} \end{bmatrix} \begin{bmatrix} 1 & -\frac{\Sigma_{12}}{\Sigma_{22}} \\ 0 & 1 \end{bmatrix}$$

Secondly, the conditional density $f(Z_1 \mid Z_2 = z_2)$ can be derived from the joint density by recognizing the quadratic form and isolating the terms that depend on Z_1. The conditional density $f(Z_1 \mid Z_2 = z_2)$ can be derived from the joint density by recognizing the quadratic form and isolating the terms that depend on Z_1. Note that $f(Z_1, Z_2) = \frac{1}{2\pi|\Sigma_Z|^{1/2}} \exp\left(-\frac{1}{2}[Z_1, Z_2]^T \Sigma_Z^{-1} [Z_1, Z_2]\right)$

Thirdly, we can break down the quadratic form: $\begin{bmatrix} Z_1 \\ Z_2 \end{bmatrix}^T \Sigma_Z^{-1} \begin{bmatrix} Z_1 \\ Z_2 \end{bmatrix}$. By expanding this product and focusing on the terms involving Z_1, we find that the quadratic form separates into two terms: $\frac{(Z_1 - \mu_{Z_1|Z_2=z_2})^2}{\Sigma_{11} - \frac{\Sigma_{12}^2}{\Sigma_{22}}} + \frac{z_2^2}{\Sigma_{22}}$, where $\mu_{Z_1|Z_2=z_2} = \frac{\Sigma_{12}}{\Sigma_{22}} z_2$.

Fourthly, From this expansion, we can now identify the conditional distribution of $Z_1 \mid Z_2 = z_2$:
Conditional Mean: $\mu_{Z_1|Z_2=z_2} = \frac{\Sigma_{12}}{\Sigma_{22}} z_2$; Conditional Variance: $\mathrm{Var}(Z_1 \mid Z_2) = \Sigma_{11} - \frac{\Sigma_{12}^2}{\Sigma_{22}}$

Thus, the conditional distribution $Z_1 \mid Z_2 = z_2$ is:

$$Z_1 \mid Z_2 = z_2 \sim \mathcal{N}\left(\frac{\Sigma_{12}}{\Sigma_{22}} z_2, \Sigma_{11} - \frac{\Sigma_{12}^2}{\Sigma_{22}}\right)$$

2 Projections and Linear Regression

We are given $X \in \mathbb{R}^{n \times d}$ where $n > d$ and $\text{rank}(X) = d$. We are also given a vector $y \in \mathbb{R}^n$. Define the orthogonal projection of y onto $\text{range}(X)$ as $P_{\text{range}(X)}(y)$.

Background on orthogonal projections For any finite-dimensional subspace W (here, $\text{range}(X)$) of a vector space V (here, $\mathbb{R}^n$), any vector $v \in V$ can be decomposed as

$$v = w + u, \qquad w \in W, \quad u \in W^{\perp},$$

where $W^{\perp}$ is the orthogonal complement of W. Furthermore, this decomposition is unique: if $v = w' + u'$ where $w' \in W$, $u' \in W^{\perp}$, then $w' = w$ and $u' = u$. These two facts allow us to define P_W, the orthogonal projection operator onto W. Given a vector v with decomposition $v = w + u$, we define

$$P_W(v) = w.$$

It can also be shown using these two facts that P_W is linear. For more information on orthogonal projections, see `https://gwthomas.github.io/docs/math4ml.pdf`.

(a) Prove that $P_{\text{range}(X)}(y) = \underset{w \in \text{range}(X)}{\arg\min} \|y - w\|_2^2$.

Solution:
Firstly, it is implied that $u = y - w$ is orthogonal to the subspace $\text{range}(X)$, which means: $\langle u, v \rangle = 0$ for all $v \in \text{range}(X)$, or equivalently: $\langle y - w, v \rangle = 0$ for all $v \in \text{range}(X)$. This is the defining property of the projection: the difference $y - w$ must be orthogonal to the subspace onto which we are projecting.

Secondly, the objective function $f(w) = \|y - w\|_2^2$, which represents the squared Euclidean distance between y and a vector w in $\text{range}(X)$, has to be minimized, in order to find the vector $w \in \text{range}(X)$ that is closest to y, by definition of orthogonal projection. We start with the objective function: $f(w) = \|y - w\|_2^2 = (y - w)^T (y - w)$. Expanding the squared norm: $f(w) = y^T y - 2y^T w + w^T w$. To minimize this expression, we take the derivative with respect to w and set it to zero: $\nabla_w f(w) = -2y + 2w = 0$. This gives: $w = P_{\text{range}(X)}(y)$. Thus, the projection w that satisfies this condition is the one that minimizes $\|y - w\|_2^2$.

Thus, the orthogonal projection of y onto $\text{range}(X)$, denoted by $P_{\text{range}(X)}(y)$, is the vector $w \in \text{range}(X)$ that minimizes the squared distance $\|y - w\|_2^2$. Therefore, we have shown that: $P_{\text{range}(X)}(y) = \arg\min_{w \in \text{range}(X)} \|y - w\|_2^2$. This completes the proof.

(b) An orthogonal projection is a linear transformation. That is, $P_{\mathrm{range}(X)}(y) = Py$ for some projection matrix P. Specifically, given $1 \leq d \leq n$, a matrix $P \in \mathbb{R}^{n \times n}$ is said to be a rank-d orthogonal projection matrix if

- $\mathrm{rank}(P) = d$
- $P = P^T$
- $P^2 = P$.

Prove that P is a rank-d projection matrix if and only if there exists a $U \in \mathbb{R}^{n \times d}$ such that $P = UU^\top$ and $U^\top U = I$.

Hint Use the eigendecomposition of P to prove the forward direction.

Solution:

To prove the forward direction (Sufficiency): We are given that P is a rank-d projection matrix and need to prove that $P = UU^T$ where $U^T U = I$.

Firstly, we use the Eigendecomposition of P. Since P is a symmetric matrix (i.e., $P = P^T$), it has an eigendecomposition. By the Spectral Theorem, any symmetric matrix can be diagonalized by an orthonormal set of eigenvectors. Let $\lambda_1, \lambda_2, \ldots, \lambda_n$ be the eigenvalues of P, and let $v_1, v_2, \ldots, v_n$ be the corresponding orthonormal eigenvectors. Since $P^2 = P$, the eigenvalues λ_i must satisfy $\lambda_i^2 = \lambda_i$. This implies that the eigenvalues of P must be either 0 or 1. Thus, the eigendecomposition of P is: $P = V \Lambda V^T$ where: $V \in \mathbb{R}^{n \times n}$ is the matrix of eigenvectors (an orthogonal matrix, so $V^T V = I$), and $\Lambda \in \mathbb{R}^{n \times n}$ is the diagonal matrix with diagonal entries $\lambda_i \in \{0, 1\}$. Because $\mathrm{rank}(P) = d$, there are exactly d eigenvalues equal to 1 and $n - d$ eigenvalues equal to 0. Therefore, we can write Λ as: $\Lambda = \begin{bmatrix} I_d & 0 \\ 0 & 0_{n-d} \end{bmatrix}$ where I_d is the $d \times d$ identity matrix and 0_{n-d} is a zero matrix. Thus, the matrix P becomes: $P = V \begin{bmatrix} I_d & 0 \\ 0 & 0_{n-d} \end{bmatrix} V^T$.

Secondly, we construct U. Let $U \in \mathbb{R}^{n \times d}$ be the first d columns of V. Since V is an orthogonal matrix, the columns of U are orthonormal, and $U^T U = I_d$. Then, we can write P as: $P = UU^T$ where $U \in \mathbb{R}^{n \times d}$ and $U^T U = I_d$.

Thus, we have shown that if P is a rank-d orthogonal projection matrix, then there exists a matrix $U \in \mathbb{R}^{n \times d}$ such that $P = UU^T$ and $U^T U = I_d$.

To prove the reverse direction (Necessity), we need to prove that if $P = UU^T$ where $U \in \mathbb{R}^{n \times d}$ and $U^T U = I_d$, then P is a rank-d orthogonal projection matrix.

Firstly, for symmetricity. $P = UU^T$ is symmetric because: $P^T = (UU^T)^T = UU^T = P$. So, $P = P^T$.

Secondly, for Idempotence, $P^2 = P$: $P^2 = (UU^T)(UU^T) = U(U^T U)U^T = UI_d U^T = UU^T = P$. Thus, $P^2 = P$, so P is idempotent.

Thirdly, for rank. The rank of $P = UU^T$ is d because $U \in \mathbb{R}^{n \times d}$, and the columns of U are linearly independent (since $U^T U = I_d$).

Thus, P is a rank-d orthogonal projection matrix.

(c) The Singular Value Decomposition theorem states that we can write any matrix X as

$$X = \sum_{i=1}^{\min\{n,d\}} \sigma_i u_i v_i^\top = \sum_{i:\sigma_i>0} \sigma_i u_i v_i^\top$$

where $\sigma_i \geq 0$, and $\{u_i\}_{i=1}^n$ and $\{v_i\}_{i=1}^d$ are orthonormal bases for $\mathbb{R}^n$ and $\mathbb{R}^d$ respectively. Some of the singular values σ_i may equal 0, indicating that the associated left and right singular vectors u_i and v_i do not contribute to the sum, but sometimes it is still convenient to include them in the SVD so we have complete orthonormal bases for $\mathbb{R}^n$ and $\mathbb{R}^d$ to work with. Show that

(i) $\{u_i : \sigma_i > 0\}$ is an orthonormal basis for the columnspace of X

(ii) Similarly, $\{v_i : \sigma_i > 0\}$ is an orthonormal basis for the row space of of X
Hint: consider $X^\top$*.*

Solution:

(i)
To show $\{u_i : \sigma_i > 0\}$ is an orthonormal basis for the columnspace of X, we observe that the SCD tells us that the matrix X can be written as a sum of rank-1 matrices $\sigma_i u_i v_i^T$, where each term $\sigma_i u_i v_i^T$ is a rank-1 contribution corresponding to the i-th singular value. More importantly, for the column space, we are interested in the left singular vectors u_i, because they span the space in $\mathbb{R}^n$ where the columns of X live.

Firstly, to show Orthonormality. The left singular vectors u_i come from the eigenvectors of the matrix XX^T, which is a symmetric matrix. The eigenvectors of symmetric matrices are orthonormal, which means: $u_i^T u_j = \delta_{ij}$ (1 if $i = j$, 0 otherwise). Therefore, the vectors $u_1, u_2, \ldots, u_{\min(n,d)}$ are orthonormal.

Secondly, to show Basis for Column Space. The vectors u_i for which $\sigma_i > 0$ correspond to non-zero singular values. These singular values capture the "directions" in which X has non-zero action. Thus, the set $\{u_i : \sigma_i > 0\}$ spans the column space of X. The number of non-zero singular values is equal to the rank of X, which defines the dimension of the column space.

(ii)

To show $\{u_i : \sigma_i > 0\}$ is an orthonormal basis for the rowspace of X, we observe that the matrix X can be written as a sum of outer products $u_i v_i^T$. For the row space, the right singular vectors $v_i \in \mathbb{R}^d$ determine the directions in which the rows of X span the space.

Firstly, to show Orthonormality. The right singular vectors v_i come from the eigenvectors of the matrix $X^T X$, which is also symmetric. The eigenvectors of symmetric matrices are orthonormal, so: $v_i^T v_j = \delta_{ij}$ (Kronecker delta: 1 if $i = j$, 0 otherwise). Therefore, the vectors $v_1, v_2, \ldots, v_{\min(n,d)}$ are orthonormal.

Secondly, to show Basis for Row Space. The vectors v_i for which $\sigma_i > 0$ correspond to the non-zero singular values. These vectors span the row space of X, as the right singular vectors give the directions in which X has non-zero action along the rows.

(d) Let $X \in \mathbb{R}^{n \times d}$ such that $\operatorname{rank}(X) = d$. Prove that $X(X^T X)^{-1} X^T$ is a rank-d orthogonal projection matrix.

Hint: Consider the SVD decomposition of X.

Solution:

Firstly, we observe several basic information. The matrix $X \in \mathbb{R}^{n \times d}$ is full rank ($\operatorname{rank}(X) = d$). The term $X^T X$ is a $d \times d$ matrix, and because X has full column rank ($d \leq n$), $X^T X$ is invertible. Therefore, $(X^T X)^{-1}$ exists. The matrix $X(X^T X)^{-1} X^T \in \mathbb{R}^{n \times n}$.

Secondly, we have to check the oroperties of a Projection Matrix. For $X(X^T X)^{-1} X^T$ to be a projection matrix, it must satisfy the following properties: - Symmetry: $P = P^T$ - Idempotence: $P^2 = P$ - Rank: $\operatorname{rank}(P) = d$

Thirdly, to proving $X(X^T X)^{-1} X^T$ is symmetric: $(X(X^T X)^{-1} X^T)^T = X(X^T X)^{-1} X^T$. This holds because: $(X^T)^T = X$, and $((X^T X)^{-1})^T = (X^T X)^{-1}$ (since $X^T X$ is symmetric and invertible).

Fourth, to prove Idempotence: $P^2 = P$, i.e., $X(X^T X)^{-1} X^T$ is idempotent: $P^2 = (X(X^T X)^{-1} X^T)(X(X^T X)^{-1} X^T)$. This simplifies to: $P^2 = X(X^T X)^{-1} X^T X (X^T X)^{-1} X^T$. Notice that $X^T X$ is a square $d \times d$ invertible matrix, so: $X^T X (X^T X)^{-1} = I_d$. Thus: $P^2 = X(X^T X)^{-1} X^T = P$.

Fifth, to prove that $\operatorname{rank}(P) = d$. The matrix $P = X(X^T X)^{-1} X^T$ projects vectors onto the column space of X. The column space of X has dimension d, since $\operatorname{rank}(X) = d$. Therefore, $\operatorname{rank}(P) = d$, as P projects onto a subspace of dimension d.

Thus, $X(X^T X)^{-1} X^T$ is a rank-d orthogonal projection matrix.

(e) Prove that $X(X^TX)^{-1}X^T$ is a projection onto range(X).

Solution:

Firstly, we prove that P acts as the identity on vectors in the range of X (i.e., for any vector $y \in \text{range}(X)$, $Py = y$). Let $y \in \text{range}(X)$. This means that there exists some vector $\alpha \in \mathbb{R}^d$ such that: $y = X\alpha$. Now, apply $P = X(X^TX)^{-1}X^T$ to y: $Py = X(X^TX)^{-1}X^Ty = X(X^TX)^{-1}X^T(X\alpha)$. Since X^TX is invertible (due to the full rank assumption of X), we can simplify: $Py = X(X^TX)^{-1}(X^TX)\alpha = X\alpha = y$. Thus, $Py = y$ for any $y \in \text{range}(X)$.

Secondly, we prove that P annihilates vectors orthogonal to range(X) (i.e., for any vector $z \in \text{range}(X)^\perp$, $Pz = 0$). Let $z \in \text{range}(X)^\perp$, meaning that z is orthogonal to the columns of X. This implies that $X^Tz = 0$. Now, apply $P = X(X^TX)^{-1}X^T$ to z: $Pz = X(X^TX)^{-1}X^Tz$. Since $X^Tz = 0$, we have: $Pz = X(X^TX)^{-1}0 = X0 = 0$. Thus, $Pz = 0$ for any $z \in \text{range}(X)^\perp$.

These two properties will confirm that $P = X(X^TX)^{-1}X^T$ is the projection matrix onto range(X).

(f) Show that $w^* = (X^TX)^{-1}X^Ty$ is the solution to the optimization problem

$$\arg\min_w \|y - Xw\|_2^2$$

using only facts proved in this problem.

Solution:

Firstly, $f(w) = \|y-Xw\|_2^2$ can be expanded as follows: $f(w) = (y-Xw)^T(y-Xw)$. Expanding this expression: $f(w) = y^Ty - 2y^TXw + w^TX^TXw$.This is a quadratic function in w, where: y^Ty is a constant, $-2y^TXw$ is linear in w, w^TX^TXw is quadratic in w.

Secondly, to find the minimum of $f(w)$, we take the gradient with respect to w and set it equal to zero: $\nabla_w f(w) = -2X^Ty + 2X^TXw = 0$. Simplifying this equation: $X^TXw = X^Ty$.

Thirdly, since X^TX is invertible (because $\text{rank}(X) = d$), we can solve for w: $w = (X^TX)^{-1}X^Ty$.

Thus, the optimal solution to the least-squares problem is: $w^* = (X^TX)^{-1}X^Ty$.

3 Some MLEs

For this question, assume you observe n (data point, label) pairs $(x_i, y_i)_{i=1}^n$, with $x_i \in \mathbb{R}^d$ and $y_i \in \mathbb{R}$ for all $i = 1, \dots, n$. We denote X as the data matrix containing all the data points and y as the label vector containing all the labels:

$$X = \begin{bmatrix} x_1^\top \\ \vdots \\ x_n^\top \end{bmatrix} \in \mathbb{R}^{n \times d}, \qquad y = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \in \mathbb{R}^n.$$

(a) Ignoring y for now, suppose we model the data points as coming from a d-dimensional Gaussian with diagonal covariance:

$$\forall i = 1, \dots, n, \quad x_i \overset{i.i.d.}{\sim} N(\mu, \Sigma); \quad \Sigma = \begin{bmatrix} \sigma_1^2 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \sigma_d^2 \end{bmatrix}.$$

If we consider $\mu \in \mathbb{R}^d$ and $(\sigma_1^2, \dots, \sigma_d^2)$, where each $\sigma_i^2 > 0$, to be unknown, the parameter space here is $2d$-dimensional. When we refer to Σ as a parameter, we are referring to the d-tuple $(\sigma_1^2, \dots, \sigma_d^2)$, but inside a linear algebraic expression, Σ denotes the diagonal matrix $\text{diag}(\sigma_1^2, \dots, \sigma_d^2)$.

Solve the following problems:

(i) Prove that log-likelihood $\ell(\mu, \Sigma) = \log p(X \mid \mu, \Sigma)$ is equal to

$$-\frac{n}{2}\left(d\log(2\pi) - \sum_{j=1}^{d} \log\left(\frac{1}{\sigma_j^2}\right)\right) - \frac{1}{2}\sum_{i=1}^{n}(x_i - \mu)^T \Sigma^{-1}(x_i - \mu).$$

(ii) Find the MLE of μ assuming Σ is known.

(iii) Find the MLE of Σ assuming μ is known.
Hint: you can re-parameterize σ_j^2 by defining $v_j = \frac{1}{\sigma_j^2}$

(iv) Find the joint MLE of (μ, Σ) in terms of the maximum likelihood estimates computed above.

Solution:

(i)

Firstly, we want to compute the Log-Likelihood for n i.i.d. Data Points.

The likelihood function for the data points $x_1, x_2, \dots, x_n$ is:

$$p(X \mid \mu, \Sigma) = \prod_{i=1}^{n} p(x_i \mid \mu, \Sigma)$$

Taking the log of the likelihood gives the log-likelihood:

$$\ell(\mu, \Sigma) = \log p(X \mid \mu, \Sigma) = \sum_{i=1}^{n} \log p(x_i \mid \mu, \Sigma)$$

where the probability density function for a multivariate normal distribution $\mathcal{N}(\mu, \Sigma)$ is given by:

$$p(x \mid \mu, \Sigma) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x - \mu)^T \Sigma^{-1}(x - \mu)\right)$$

For each data point x_i, the log of the probability density is:

$$\log p(x_i \mid \mu, \Sigma) = -\frac{d}{2}\log(2\pi) - \frac{1}{2}\log|\Sigma| - \frac{1}{2}(x_i - \mu)^T\Sigma^{-1}(x_i - \mu)$$

where Σ is diagonal, which means the determinant $|\Sigma|$ is:

$$|\Sigma| = \prod_{j=1}^{d}\sigma_j^2$$

Substituting $|\Sigma| = \prod_{j=1}^{d}\sigma_j^2$, we get:

$$\log p(x_i \mid \mu, \Sigma) = -\frac{d}{2}\log(2\pi) - \frac{1}{2}\sum_{j=1}^{d}\log(\sigma_j^2) - \frac{1}{2}(x_i - \mu)^T\Sigma^{-1}(x_i - \mu)$$

Secondly, we want to sum over all n data points, we have:

$$\ell(\mu, \Sigma) = \sum_{i=1}^{n}\left(-\frac{d}{2}\log(2\pi) - \frac{1}{2}\sum_{j=1}^{d}\log(\sigma_j^2) - \frac{1}{2}(x_i - \mu)^T\Sigma^{-1}(x_i - \mu)\right)$$

Simplifying:

$$\ell(\mu, \Sigma) = -\frac{nd}{2}\log(2\pi) - \frac{n}{2}\sum_{j=1}^{d}\log(\sigma_j^2) - \frac{1}{2}\sum_{i=1}^{n}(x_i - \mu)^T\Sigma^{-1}(x_i - \mu)$$

(ii.)

Firstly, we want to find the Log-likelihood Function for μ. Since Σ is known, we ignore the terms that don't involve μ, and focus on the second part of the log-likelihood of $\ell(\mu, \Sigma) = -\frac{n}{2}\left(d\log(2\pi) + \sum_{j=1}^{d}\log(\sigma_j^2)\right) - \frac{1}{2}\sum_{i=1}^{n}(x_i - \mu)^T\Sigma^{-1}(x_i - \mu)$. This gives us $\ell(\mu) = -\frac{1}{2}\sum_{i=1}^{n}(x_i - \mu)^T\Sigma^{-1}(x_i - \mu)$

Secondly, to Maximize the Log-likelihood, we want to find the MLE of μ, which we take the gradient of $\ell(\mu)$ with respect to μ and set it equal to zero:$\nabla_\mu\ell(\mu) = -\frac{1}{2}\nabla_\mu\left(\sum_{i=1}^{n}(x_i - \mu)^T\Sigma^{-1}(x_i - \mu)\right)$. Taking the derivative with respect to μ: $\nabla_\mu\ell(\mu) = \Sigma^{-1}\sum_{i=1}^{n}(x_i - \mu)$. Set the gradient equal to zero: $\Sigma^{-1}\sum_{i=1}^{n}(x_i - \mu) = 0$. This simplifies to:$\sum_{i=1}^{n}(x_i - \mu) = 0$. Rearanging gives :$n\mu = \sum_{i=1}^{n}x_i$

Finally, solving for μ, we get the MLE:

$$\boxed{\mu^* = \frac{1}{n}\sum_{i=1}^{n}x_i}$$

(iii.)

Firstly, we want to reformat the expression we obtained from (i.). Since Σ^{-1} is diagonal, we have: $\Sigma^{-1} = \text{diag}\left(\frac{1}{\sigma_1^2}, \frac{1}{\sigma_2^2}, \ldots, \frac{1}{\sigma_d^2}\right)$ This means the quadratic form $(x_i - \mu)^T\Sigma^{-1}(x_i - \mu)$ simplifies to: $(x_i - \mu)^T\Sigma^{-1}(x_i - \mu) = \sum_{j=1}^{d}\frac{(x_{ij} - \mu_j)^2}{\sigma_j^2}$ Therefore, the log-likelihood becomes:

$$\ell(\mu, \Sigma) = -\frac{n}{2}\left(d\log(2\pi) + \sum_{j=1}^{d}\log(\sigma_j^2)\right) - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{d}\frac{(x_{ij} - \mu_j)^2}{\sigma_j^2}$$

Secondly, we would reparameterize the problem. We are given the hint to re-parameterize σ_j^2 by defining $\nu_j = \frac{1}{\sigma_j^2}$. Substituting this into the log-likelihood:

$$\ell(\nu) = -\frac{n}{2}\left(d\log(2\pi) - \sum_{j=1}^{d}\log(\nu_j)\right) - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{d}\nu_j(x_{ij}-\mu_j)^2$$

Thirdly, we want to maximize the log-likelihood. To find the MLE, we take the derivative of $\ell(\nu)$ with respect to ν_j and set it equal to zero: $\frac{\partial\ell(\nu)}{\partial\nu_j} = \frac{n}{2\nu_j} - \frac{1}{2}\sum_{i=1}^{n}(x_{ij}-\mu_j)^2 = 0$ This simplifies to: $\frac{n}{\nu_j} = \sum_{i=1}^{n}(x_{ij}-\mu_j)^2$ Thus, solving for ν_j: $\nu_j = \frac{n}{\sum_{i=1}^{n}(x_{ij}-\mu_j)^2}$ Since $\nu_j = \frac{1}{\sigma_j^2}$, we have:

$$\sigma_j^2 = \frac{1}{\nu_j} = \frac{1}{\frac{n}{\sum_{i=1}^{n}(x_{ij}-\mu_j)^2}} = \frac{1}{n}\sum_{i=1}^{n}(x_{ij}-\mu_j)^2$$

Thus, the MLE of σ_j^2 is:

$$\boxed{\sigma_j^{2*} = \frac{1}{n}\sum_{i=1}^{n}(x_{ij}-\mu_j)^2}$$

(iv.)

The joint MLE of (μ, Σ) is simply the combination of the individual MLEs for μ and Σ.

Thus, the joint MLE of (μ, Σ) is:

$$\boxed{(\mu^*, \Sigma^*) = \left(\frac{1}{n}\sum_{i=1}^{n}x_i,\ \mathrm{diag}\left(\frac{1}{n}\sum_{i=1}^{n}(x_{i1}-\mu_1)^2, \dots, \frac{1}{n}\sum_{i=1}^{n}(x_{id}-\mu_d)^2\right)\right)}$$

where $\mu^* = \frac{1}{n}\sum_{i=1}^{n}x_i$ is the sample mean, and Σ^* is the diagonal matrix with elements $\sigma_j^{2*} = \frac{1}{n}\sum_{i=1}^{n}(x_{ij}-\mu_j)^2$ for $j = 1, \dots, d$.

(b) Suppose that we have a training set $\{(x_i, y_i) \mid i = 1 \ldots n\}$ of n independent examples but in which the residual terms had different variances. That is, we assume

$$y_i \sim N(w^T x_i, \sigma_i^2).$$

Show that the MLE estimate of w can be found by solving the following optimization problem

$$w_{\text{MLE}} = \arg\min_w \|A(Xw - y)\|_2^2.$$

Clearly state what the matrix A equals.

Solution:

Firstly, we want to write the log-likihood function. The likelihood of the data is: $L(w) = \prod_{i=1}^n \frac{1}{\sqrt{2\pi\sigma_i^2}} \exp\left(-\frac{(y_i - w^T x_i)^2}{2\sigma_i^2}\right)$ The log-likelihood is: $\ell(w) = \sum_{i=1}^n \left(-\frac{1}{2}\log(2\pi\sigma_i^2) - \frac{(y_i - w^T x_i)^2}{2\sigma_i^2}\right)$ Since the first term does not depend on w, we minimize the second term:

$$\ell(w) = -\frac{1}{2}\sum_{i=1}^n \frac{(y_i - w^T x_i)^2}{\sigma_i^2}$$

Secondly, we express this in matrix form. Let $X \in \mathbb{R}^{n \times d}$ be the matrix of predictors, $y \in \mathbb{R}^n$ the response vector, and $w \in \mathbb{R}^d$ the parameter vector. The log-likelihood becomes:

$$\ell(w) = -\frac{1}{2}(y - Xw)^T \Sigma^{-1} (y - Xw)$$

where $\Sigma = \text{diag}(\sigma_1^2, \sigma_2^2, \ldots, \sigma_n^2)$.

Thirdly, we solve the optimization problem. Let $A = \Sigma^{-1/2}$, i.e., $A = \text{diag}\left(\frac{1}{\sigma_1}, \frac{1}{\sigma_2}, \ldots, \frac{1}{\sigma_n}\right)$. Then the negative log-likelihood becomes: $\|A(Xw - y)\|_2^2$

Thus, the MLE is found by solving:

$$w_{\text{MLE}} = \arg\min_w \|A(Xw - y)\|_2^2$$

where $A = \text{diag}\left(\frac{1}{\sigma_1}, \frac{1}{\sigma_2}, \ldots, \frac{1}{\sigma_n}\right)$.

(c) Consider the Categorical$(\theta_1, \theta_2, \dots, \theta_k)$ distribution. Recall, for categorical distributions, there are two constraints on θ_k:

- $\theta_k \geq 0$ for all k
- $\sum_{k=1}^{K} \theta_k = 1$

The distribution describes a random process that selects one of the K possible categories, with category k being chosen with probability θ_k.

Ignoring the data points X, suppose that for all i from 1 to n, we sample y_i from a categorical distribution:

$$y_i \overset{i.i.d.}{\sim} \text{Categorical}(\theta_1, \dots, \theta_K).$$

Compute the MLE of $\theta = (\theta_1, \dots, \theta_K)$. Use the fact that the KL divergence is nonnegative:

$$\text{KL}(\pi \,\|\, \theta) = \sum_{\omega \in \Omega} \pi(\omega) \log \left(\frac{\pi(\omega)}{\theta(\omega)} \right) \geq 0.$$

Solution:

Firstly, we derive the likelihood function. Let n_k represent the number of times category k appears in the data. The likelihood function is: $L(\theta) = \prod_{k=1}^{K} \theta_k^{n_k}$ where n_k is the count of how often category k was observed, and $\sum_{k=1}^{K} n_k = n$. We can then derive the log-likelihood as:

$$\ell(\theta) = \log L(\theta) = \sum_{k=1}^{K} n_k \log \theta_k$$

Secondly, we maximize $\ell(\theta)$ subject to the constraint $\sum_{k=1}^{K} \theta_k = 1$.We use Lagrange multipliers, defining the Lagrange function: $\mathcal{L}(\theta, \lambda) = \sum_{k=1}^{K} n_k \log \theta_k + \lambda \left(1 - \sum_{k=1}^{K} \theta_k\right)$. We take the derivative of $\mathcal{L}$ with respect to θ_k and setting it equal to zero: $\frac{\partial}{\partial \theta_k} \mathcal{L}(\theta, \lambda) = \frac{n_k}{\theta_k} - \lambda = 0$. Solving for θ_k: $\theta_k = \frac{n_k}{\lambda}$. We enforce the constraint $\sum_{k=1}^{K} \theta_k = 1$:

$$\sum_{k=1}^{K} \frac{n_k}{\lambda} = 1 \quad \Rightarrow \quad \lambda = n$$

Thus, the MLE of $\theta = (\theta_1, \dots, \theta_K)$ is:

$$\boxed{\theta_k^* = \frac{n_k}{n}}$$

(d) Again consider X fixed. This time, we suppose that each y_i is binary-valued (0 or 1). We choose to model y as

$$y_i \overset{ind.}{\sim} \text{Ber}(s(x_i^\top w)) \quad \forall i = 1, \dots, n,$$

where $s(z) = \frac{1}{1+e^{-z}}$ is the *sigmoid* function and $\text{Ber}(p)$ denotes the Bernoulli distribution which takes value 1 with probability p and 0 with probability $1-p$.

(i) Write down the log-likelihood $\ell(w) = \log p(y \,|\, w)$ and show that finding the MLE of w is equivalent to minimizing the cross entropy between $\text{Ber}(y_i)$ and $\text{Ber}(s(x_i^\top w))$ for each i:

$$\min_{w \in \mathbb{R}^d} \sum_{i=1}^{n} H(\text{Ber}(y_i), \text{Ber}(s(x_i^\top w))). \tag{1}$$

Definition of cross entropy: given two discrete probability distributions $\pi : \Omega \to [0,1]$ and $\theta : \Omega \to [0,1]$ on some outcome space Ω, we define the cross entropy between π and θ as

$$H(\pi, \theta) = \sum_{\omega \in \Omega} -\pi(\omega) \log \theta(\omega).$$

Solution:

Firstly, The likelihood of the data is: $L(w) = \prod_{i=1}^{n} s(x_i^T w)^{y_i} (1 - s(x_i^T w))^{1-y_i}$ Taking the log of the likelihood gives the log-likelihood:

$$\ell(w) = \log L(w) = \sum_{i=1}^{n} \left[y_i \log s(x_i^T w) + (1-y_i) \log(1 - s(x_i^T w)) \right]$$

Secondly, the cross-entropy between two Bernoulli distributions $\text{Ber}(y_i)$ and $\text{Ber}(s(x_i^T w))$ is: $H(\text{Ber}(y_i), \text{Ber}(s(x_i^T w))) = - \left[y_i \log s(x_i^T w) + (1-y_i) \log(1 - s(x_i^T w)) \right]$ Summing over all data points:

$$\sum_{i=1}^{n} H(\text{Ber}(y_i), \text{Ber}(s(x_i^T w))) = - \sum_{i=1}^{n} \left[y_i \log s(x_i^T w) + (1-y_i) \log(1 - s(x_i^T w)) \right]$$

Thirdly, Maximizing the log-likelihood is equivalent to minimizing the negative log-likelihood:

$$\min_w -\ell(w) = \min_w \sum_{i=1}^{n} \left[-y_i \log s(x_i^T w) - (1-y_i) \log(1 - s(x_i^T w)) \right]$$

Thus, upon comparison, finding the MLE of w is equivalent to minimizing the cross-entropy:

$$\min_w \sum_{i=1}^{n} H(\text{Ber}(y_i), \text{Ber}(s(x_i^T w)))$$

(ii) Show that (1) (and therefore finding the MLE) is equivalent to the following problem:

$$\min_{w \in \mathbb{R}^d} \sum_{i=1}^{n} \log(1 + \exp(-z_i x_i^\top w)) \tag{2}$$

where $z_i = 1$ if $y_i = 1$ and $z_i = -1$ if $y_i = 0$.

Note: both (1) and (2) are referred to as logistic regression.

Solution:

Firstly, we perform Cross-Entropy Loss for Logistic Regression. From part (i), we derived the log-likelihood: $\ell(w) = \sum_{i=1}^{n} \left[y_i \log s(x_i^T w) + (1-y_i) \log(1 - s(x_i^T w)) \right]$ where $s(x_i^T w) = \frac{1}{1+e^{-x_i^T w}}$ is the sigmoid function.

The negative log-likelihood (which we want to minimize) is:

$$-\ell(w) = \sum_{i=1}^{n} \left[-y_i \log s(x_i^T w) - (1 - y_i) \log(1 - s(x_i^T w))\right]$$

Secondly, we proceed with prove by cases:
Logistic Loss for $y_i = 1$: For $y_i = 1$, the cross-entropy term becomes:

$$-\log s(x_i^T w) = \log(1 + e^{-x_i^T w})$$

This is the logistic loss for $z_i = 1$:

$$\log(1 + \exp(-z_i x_i^T w)) = \log(1 + \exp(-x_i^T w)) \quad \text{for } z_i = 1.$$

Logistic Loss for $y_i = 0$:
For $y_i = 0$, the cross-entropy term becomes:

$$-\log(1 - s(x_i^T w)) = \log(1 + e^{x_i^T w})$$

This is the logistic loss for $z_i = -1$:

$$\log(1 + \exp(-z_i x_i^T w)) = \log(1 + \exp(x_i^T w)) \quad \text{for } z_i = -1.$$

Thus, the cross-entropy loss is equivalent to the logistic loss:

$$\sum_{i=1}^{n} H(\text{Ber}(y_i), \text{Ber}(s(x_i^T w))) = \sum_{i=1}^{n} \log(1 + \exp(-z_i x_i^T w))$$

Thus, equation (1) and equation (2) are equivalent.

(iii) Let $J(w) = \log(1+\exp(-zx^\top w))$ where, again, $z = 1$ if $y = 1$ and $z = -1$ if $y = 0$ (we are only considering a single (x, y) pair in this subpart). Prove the following:

i. J is not strictly convex.
Hint: A necessary condition for a twice-differentiable function to be strictly convex is that its Hessian is positive definite.

ii. The gradient descent update rule for minimizing $J(w)$ with learning rate ϵ is

$$w' = w - \epsilon \left(\frac{1}{1 + e^{-x^T w}} - y \right) x$$

Solution:
To prove that (i) J is not strictly convex, we observe that a necessary condition for strict convexity is that the Hessian of the function is positive definite.
Firstly, we compute the Gradient of $J(w)$. The function is: $J(w) = \log(1 + \exp(-zx^T w))$ Taking the gradient:

$$\nabla J(w) = \frac{-zx}{1 + \exp(zx^T w)}$$

Secondly, we compute the Hessian. The Hessian is the second derivative of $J(w)$ with respect to w: $\nabla^2 J(w) = \frac{\exp(zx^T w) z^2 x x^T}{(1+\exp(zx^T w))^2}$ Since $z^2 = 1$, this simplifies to:

$$\nabla^2 J(w) = \frac{\exp(zx^T w) x x^T}{(1 + \exp(zx^T w))^2}$$

Thirdly, we analyze the Hessian. The Hessian is a rank-1 matrix (the outer product xx^T). A rank-1 matrix has a single positive eigenvalue (in the direction of x) and zero eigenvalues in all other directions. Therefore, the Hessian is not positive definite, and $J(w)$ is not strictly convex.

To find (ii) Gradient Descent Update Rule, we start with the the general gradient descent update rule:

$$w' = w - \epsilon \nabla J(w)$$

Secondly, we substituted the gradient we computed from previous part: $\nabla J(w) = \frac{-zx}{1+\exp(zx^T w)}$ into the update rule: $w' = w - \epsilon \left(\frac{-zx}{1+\exp(zx^T w)} \right)$. This simplifies to:

$$w' = w + \epsilon \left(\frac{zx}{1 + \exp(zx^T w)} \right)$$

Finally, using $z = 2y - 1$, we rewrite the update rule as:

$$w' = w - \epsilon \left(\frac{1}{1 + \exp(-x^T w)} - y \right) x$$

4 Geometry of Ridge Regression

You recently learned ridge regression and how it differs from ordinary least squares. In this question we will explore the properties of ridge regression in more depth. Recall that the ridge regression problem is given by the following optimization problem:

$$\min_{w} \|\mathbf{y} - \mathbf{X}\mathbf{w}\|_2^2 + \nu\|\mathbf{w}\|_2^2. \tag{3}$$

The solution to ridge regression is given by

$$\hat{\mathbf{w}}_\mathbf{r} = (\mathbf{X}^\top\mathbf{X} + \nu\mathbf{I})^{-1}\mathbf{X}^\top\mathbf{y}. \tag{4}$$

1. One reason why we might want to have small weights $\mathbf{w}$ has to do with the sensitivity of the predictor to its input. Let $\mathbf{x}$ be a d-dimensional list of features corresponding to a new test point. Our predictor is $\mathbf{w}^\top\mathbf{x}$. What is an upper bound on how much our prediction could change if we added noise $\boldsymbol{\epsilon} \in \mathbb{R}^d$ to a test point's features $\mathbf{x}$, in terms of $\|\mathbf{w}\|_2$ and $\|\boldsymbol{\epsilon}\|_2$?
   Hint: Use the Cauchy-Schwarz inequality.

   Solution: Solution:

   Firstly, the predictor is given by $\mathbf{w}^T\mathbf{x}$, and if we add noise, the new prediction becomes: $\mathbf{w}^T(\mathbf{x}+\epsilon) = \mathbf{w}^T\mathbf{x}+\mathbf{w}^T\epsilon$ which means the change in the prediction due to the noise is: $\Delta = \mathbf{w}^T\epsilon$

   Secondly, using the Cauchy-Schwarz inequality, which states: $|\mathbf{a}^T\mathbf{b}| \leq \|\mathbf{a}\|_2\|\mathbf{b}\|_2$, we apply it to Δ with $\mathbf{a} = \mathbf{w}$ and $\mathbf{b} = \epsilon$, yielding:

   $$|\Delta| = |\mathbf{w}^T\epsilon| \leq \|\mathbf{w}\|_2\|\epsilon\|_2$$

   Thus, the upper bound on the change in the prediction is:

   $$|\mathbf{w}^T\epsilon| \leq \|\mathbf{w}\|_2\|\epsilon\|_2$$

2. Note that in computing $\hat{\mathbf{w}}_{\mathbf{r}}$, we are trying to invert the matrix $\mathbf{X}^\top \mathbf{X} + \nu \mathbf{I}$ instead of the matrix $\mathbf{X}^\top \mathbf{X}$. If $\mathbf{X}^\top \mathbf{X}$ has eigenvalues $\sigma_1^2, \ldots, \sigma_d^2$, what are the eigenvalues of $(\mathbf{X}^\top \mathbf{X} + \nu \mathbf{I})^{-1}$? Comment on why adding the regularizer term $\nu \mathbf{I}$ can improve the inversion operation numerically.

Solution:

Firstly, we compute the eigenvalues of $X^T X + \nu I$. Given that $X^T X$ has eigenvalues $\sigma_1^2, \sigma_2^2, \ldots, \sigma_d^2$, we can write the eigenvalue decomposition: $X^T X = U \Lambda U^T$ where U is an orthogonal matrix of eigenvectors, and $\Lambda = \text{diag}(\sigma_1^2, \ldots, \sigma_d^2)$ is the diagonal matrix of eigenvalues. Since νI has eigenvalues ν for all its eigenvalues, and $X^T X$ and I share the same set of eigenvectors, the eigenvalues of $X^T X + \nu I$ are:

$$\sigma_1^2 + \nu, \sigma_2^2 + \nu, \ldots, \sigma_d^2 + \nu$$

Secondly, we condier the Effect of Regularization on Numerical Stability. When inverting $X^T X$, small eigenvalues lead to a large condition number and instability. Adding νI ensures that all the eigenvalues are at least ν, improving the condition number and making the matrix easier to invert.

Thus, adding the regularization term νI:

- Increases the smallest eigenvalues, improving the condition number.

- Makes the inversion operation more stable, preventing small eigenvalues from dominating.

3. Let the number of parameters $d = 4$ and the number of datapoints $n = 6$, and let the eigenvalues of $\mathbf{X}^\top\mathbf{X}$ be given by 500, 10, 1, and 0.001. We must now choose between two regularization parameters $\nu_1 = 50$ and $\nu_2 = 0.1$. Which do you think is a better choice for this problem and why?

Solution:

Firstly, we want to compute the Eigenvalues of $X^TX + \nu I$. Adding a regularization term νI shifts each eigenvalue of X^TX by ν. Therefore, the eigenvalues of $X^TX + \nu_1 I$ and $X^TX + \nu_2 I$ are:

- For $\nu_1 = 50$:

$$500 + 50 = 550, \quad 10 + 50 = 60, \quad 1 + 50 = 51, \quad 0.001 + 50 = 50.001$$

- For $\nu_2 = 0.1$:

$$500 + 0.1 = 500.1, \quad 10 + 0.1 = 10.1, \quad 1 + 0.1 = 1.1, \quad 0.001 + 0.1 = 0.101$$

Secondlly, we consider the condition number,ie. the ratio of the largest eigenvalue to the smallest eigenvalue:

- For $\nu_1 = 50$, the condition number is:

$$\frac{550}{50.001} \approx 11$$

- For $\nu_2 = 0.1$, the condition number is:

$$\frac{500.1}{0.101} \approx 4950$$

Thus, the regularization parameter $\nu_1 = 50$ is a better choice because it results in a well-conditioned matrix with a condition number of approximately 11, making the inversion more stable. In contrast, $\nu_2 = 0.1$ results in a condition number of 4950, which is still too large and could lead to numerical instability.

4. Another advantage of ridge regression can be seen for under-determined systems. Say we have the data drawn from a $d = 5$ parameter model, but only have $n = 4$ training samples of it, i.e. $\mathbf{X} \in \mathbb{R}^{4\times5}$. Now this is clearly an underdetermined system, since $n < d$. Show that ridge regression with $\nu > 0$ results in a unique solution, whereas ordinary least squares has an infinite number of solutions.

   Hint: To make this point, it may be helpful to consider $\mathbf{w} = \mathbf{w}_0 + \mathbf{w}^*$ where $\mathbf{w}_0$ is in the null space of $\mathbf{X}$ and $\mathbf{w}^*$ is a solution.

   Solution:

   Firstly, in OLS, we minimize: $\min_{\mathbf{w}} \|X\mathbf{w} - \mathbf{y}\|_2^2$ Since $X \in \mathbb{R}^{4\times5}$, the system is underdetermined. Therefore, there are infinitely many solutions because for any particular solution $\mathbf{w}^*$, we can add any vector $\mathbf{w}_0$ in the null space of X (i.e., $X\mathbf{w}_0 = 0$) without changing the objective. In other words, the solution to OLS is not unique.

   Secondly, in ridge regression, we minimize: $\min_{\mathbf{w}} \|X\mathbf{w} - \mathbf{y}\|_2^2 + \nu\|\mathbf{w}\|_2^2$ where $\nu > 0$ is the regularization parameter. The regularization term $\nu\|\mathbf{w}\|_2^2$ forces the solution to minimize the norm of $\mathbf{w}$, eliminating the possibility of adding arbitrary vectors from the null space of X. Therefore, the solution to ridge regression is unique.

   For a more rigirous proof, condier the decomposition of $\mathbf{w}$. We can decompose $\mathbf{w}$ as: $\mathbf{w} = \mathbf{w}^* + \mathbf{w}_0$ where $\mathbf{w}^*$ is a particular solution, and $\mathbf{w}_0$ is in the null space of X. In OLS, this does not affect the objective, but in ridge regression, the term $\nu\|\mathbf{w}\|_2^2$ penalizes nonzero $\mathbf{w}_0$, forcing $\mathbf{w}_0 = 0$. Thus, OLS has an infinite number of solutions, whereas Ridge regression with $\nu > 0$ results in a unique solution because the regularization term eliminates null space solutions.

5. What will the solution to ridge regression (4) converge to if you take the limit $\nu \to 0$? Your answer should be a simple expression in terms of $\mathbf{U}, \mathbf{\Sigma}, \mathbf{V}, \mathbf{y}$, and ν where $\mathbf{X} = \mathbf{U\Sigma V}^T$ is the SVD of $\mathbf{X}$.

Solution:

Solution:

Firstly, consider the SVD of X. Let $X = U\Sigma V^T$, where: $U \in \mathbb{R}^{n\times n}$ is an orthogonal matrix of left singular vectors, $\Sigma \in \mathbb{R}^{n\times d}$ is a diagonal matrix with singular values $\sigma_1, \sigma_2, \ldots, \sigma_r$, $V \in \mathbb{R}^{d\times d}$ is an orthogonal matrix of right singular vectors. This yields $X^TX = V\Sigma^T\Sigma V^T$.

Secondly, we substitute the SVD of X^TX into the ridge regression solution: $\hat{w}_r = (V\Sigma^T\Sigma V^T + \nu I)^{-1}V\Sigma^TU^Ty$ Since $V^TV = I$, this simplifies to:

$$\hat{w}_r = V(\Sigma^T\Sigma + \nu I)^{-1}\Sigma^TU^Ty$$

Thirdly, we put the Limit in play as $\nu \to 0$. As $\nu \to 0$, the term $\Sigma^T\Sigma + \nu I$ converges to $\Sigma^T\Sigma$, so the ridge regression solution converges to: $\lim_{\nu\to 0}\hat{w}_r = V(\Sigma^T\Sigma)^{-1}\Sigma^TU^Ty$ Since $\Sigma^T\Sigma$ is a diagonal matrix with entries σ_i^2, its inverse is $\frac{1}{\sigma_i^2}$, and the solution simplifies to:

$$\lim_{\nu\to 0}\hat{w}_r = V\Sigma^{-1}U^Ty$$

This is the ordinary least squares (OLS) solution using the pseudoinverse of X.

6. Tikhonov regularization is a general term for ridge regression, where the implicit constraint set takes the form of an ellipsoid instead of a ball. In other words, we solve the optimization problem

$$\mathbf{w} = \arg\min_{\mathbf{w}} \|\mathbf{y} - \mathbf{X}\mathbf{w}\|_2^2 + \nu\|\mathbf{\Gamma}\mathbf{w}\|_2^2$$

for some full rank matrix $\mathbf{\Gamma} \in \mathbb{R}^{d\times d}$. Derive a closed form solution for $\mathbf{w}$.

Solution: Solution:

Firstly, the objective function can be written as: $J(\mathbf{w}) = \|\mathbf{y} - X\mathbf{w}\|_2^2 + \nu\|\Gamma\mathbf{w}\|_2^2$ and by expanding we have:

$$J(\mathbf{w}) = (\mathbf{y} - X\mathbf{w})^T(\mathbf{y} - X\mathbf{w}) + \nu\mathbf{w}^T\Gamma^T\Gamma\mathbf{w}$$

Secondly, to minimize $J(\mathbf{w})$, we take the gradient: $\nabla_{\mathbf{w}} J(\mathbf{w}) = -2X^T(\mathbf{y} - X\mathbf{w}) + 2\nu\Gamma^T\Gamma\mathbf{w}$ which simplifies to:

$$\nabla_{\mathbf{w}} J(\mathbf{w}) = -2X^T\mathbf{y} + 2X^TX\mathbf{w} + 2\nu\Gamma^T\Gamma\mathbf{w}$$

Thirdly, we set the gradient to zero to find the optimal $\mathbf{w}$: $-X^T\mathbf{y} + X^TX\mathbf{w} + \nu\Gamma^T\Gamma\mathbf{w} = 0$. Rearranging gives:

$$(X^TX + \nu\Gamma^T\Gamma)\mathbf{w} = X^T\mathbf{y}$$

Finally, solving for $\mathbf{w}$ gives the closed-form solution:

$$\mathbf{w} = (X^TX + \nu\Gamma^T\Gamma)^{-1}X^T\mathbf{y}$$

5 Robotic Learning of Controls from Demonstrations and Images

Huey, a home robot, is learning to retrieve objects from a cupboard. The goal is to push obstacle objects out of the way to expose a goal object. Huey's robot trainer, Anne, provides demonstrations via tele-operation. When tele-operating the robot, Anne can look at the images captured by the robot and provide controls to Huey remotely.

During a demonstration, Huey records the RGB images of the scene for each of the n timesteps, $x_1, ..., x_n$, where $x_i \in \mathbb{R}^{30\times30\times3}$ and the controls for his body for each of the n timesteps, $u_1, \ldots, u_n$, where $u_i \in \mathbb{R}^3$. The controls correspond to making small changes in the 3D pose (i.e. translation and rotation) of his body. Examples of the data are shown in the figure.

Under an assumption (sometimes called the Markovian assumption) that all that matters for the current control is the current image, Huey can try to learn a linear *policy* π (where $\pi \in \mathbb{R}^{2700\times3}$) which linearly maps image states to controls (i.e. $\pi^\top x = u$). We will now explore how Huey can recover this policy using linear regression.

Note the dimensions in this problem! Previously, you saw linear regression in problems in which the learned weight w^* was a vector and the predicted value y was a scalar. Here, we are predicting 3D controls. This means that the learned policy is a matrix. In essence, we are performing 3 regressions at the same time, one for each element of the predicted control u.

Please stick to **numpy** (and **numpy.linalg**) only for performing any computations in this assignment. We will ask that you edit the file `robotic_ridge_code.py` directly, instead of working in a Python notebook, and submit it to the Gradescope autograder after you are finished. Please don't rename the file, or change any of the function signatures!

(a) To get familiar with the structure of the data, **please visualize the 0th, 10th and 20th images in the training dataset. Also find their corresponding control vectors.**
Note: the training and testing images are currently stored as float32 numpy arrays, with pixel values in the range [0.0, 255.0]. You may have to convert to these images to the np.uint8 format to visualize them.

Solution:

Control vector for image 0: [0. -1. 0.]

Control vector for image 10: [-1. -0.45111084 -1.]

Control vector for image 20: [0. 0. 0.37368774]

(b) Load the n training examples from `x_train.p` and compose the matrix X, where $X \in \mathbb{R}^{n\times 2700}$. Note that you will need to flatten the images and reduce them to a single vector. The flattened image vector will be denoted by $\bar{x}$ (where $\bar{x} \in \mathbb{R}^{2700\times 1}$). Next, load the n examples from `y_train.p` and compose the matrix U, where $U \in \mathbb{R}^{n\times 3}$. Try to perform ordinary least squares by forming the matrix $(X^\top X)^{-1}X^\top$ for solving

$$\min_{\pi} \|X\pi - U\|_F$$

in order to learn the optimal *policy* $\pi^* \in \mathbb{R}^{2700\times 3}$. **Report what happens as you attempt to do this and explain why**.

Solution:

Error message:

Linear algebra error during OLS: Singular matrix

OLS solution could not be computed due to a singular matrix.

Explanation:

If $X^T X$ is not invertible, it means that the matrix is singular, often due to multicollinearity or insufficient training data. This is common when the number of features (2700) is greater than the number of training examples n. In cases where $n < 2700$, there are fewer data points than the dimensionality of the flattened image space, which results in an underdetermined system. This makes it impossible to uniquely solve for π using ordinary least squares.

(c) Now try to perform ridge regression:

$$\min_{\pi} \|X\pi - U\|_F^2 + \lambda\|\pi\|_F^2$$

on the dataset for regularization values $\lambda = \{0.1, 1.0, 10, 100, 1000\}$. Measure the average squared Euclidean distance for the accuracy of the policy on the training data:

$$\frac{1}{n}\sum_{i=0}^{n-1} \|\bar{x}_i^T \pi - u_i^\top\|_2^2$$

In the expression above, we are taking the ℓ_2 norm of a row vector, which here we take to mean the ℓ_2 norm of the column vector we get by transposing it. **Report the training error results for each value of λ.**

Solution:

Lambda: 0.1, Training Error: 7.112299631996796e-10

Lambda: 1.0, Training Error: 2.2959248959185347e-12

Lambda: 10.0, Training Error: 1.2596346278435236e-11

Lambda: 100.0, Training Error: 1.2556408513356372e-09

Lambda: 1000.0, Training Error: 1.2459338127401795e-07

(d) Next, we are going to try standardizing the states. For each pixel value in each data point, $\bar{x}$, perform the following operation:

$$\bar{x} \mapsto \frac{\bar{x}}{255} \times 2 - 1$$

We know that the maximum pixel value is 255, so this operation rescales the data to be in the range $[-1, 1]$. **Repeat the previous part and report the average squared training error for each value of λ.**

Solution:

Lambda: 0.1, Training Error: 3.25592934685693e-07

Lambda: 1.0, Training Error: 2.9105343334579686e-05

Lambda: 10.0, Training Error: 0.001590383483089189

Lambda: 100.0, Training Error: 0.03477312478925273

Lambda: 1000.0, Training Error: 0.25440296240987653

(e) Evaluate both *policies* (i.e. with and without standardization) on the new validation data `x_test.p` and `y_test.p` for the different values of λ. **Report the average squared Euclidean loss and qualitatively explain how changing the values of λ affects the performance in terms of bias and variance**.

Solution:

Lambda: 0.1

Non-standardized Test Error: 0.7740299206171035

Standardized Test Error: 0.8679145915518948

Lambda: 1.0

Non-standardized Test Error: 0.7740215946026836

Standardized Test Error: 0.8620856552331783

Lambda: 10.0

Non-standardized Test Error: 0.7740172691223421

Standardized Test Error: 0.8275061026236492

Lambda: 100.0

Non-standardized Test Error: 0.7739831282443682

Standardized Test Error: 0.7246529988047817

Lambda: 1000.0

Non-standardized Test Error: 0.7736442559285185

Standardized Test Error: 0.7250141742255005

Explanation: The regularization parameter λ plays a crucial role in controlling the trade-off between bias and variance in ridge regression. At low values of λ, the model fits the training data more closely, resulting in low bias but high variance, as it becomes prone to overfitting by capturing noise in the data. As λ increases, the regularization becomes stronger, which constrains the model coefficients, simplifying the model. This helps reduce variance by preventing overfitting, but at the cost of increasing bias, as the model may become too simple to capture the underlying patterns in the data. At very high values of λ, the model can become too rigid, leading to high bias and low variance, resulting in underfitting, where the model fails to capture important trends. Therefore, λ needs to be chosen carefully to strike a balance between bias and variance, with moderate values typically leading to better generalization by reducing overfitting without introducing excessive bias.

(f) To better understand how standardizing improved the loss function, we are going to evaluate the *condition number* κ of the optimization problem above, which is defined as

$$\kappa = \frac{\sigma_{\text{max}}(X^T X + \lambda I)}{\sigma_{\text{min}}(X^T X + \lambda I)}$$

or the ratio of the maximum singular value to the minimum singular value of the relevant matrix. Roughly speaking, the condition number of the optimization process measures how stable the solution will be when some error exists in the observations. More precisely, given a linear system $Ax = b$, the condition number of the matrix A is the maximum ratio of the relative error in the solution x to the relative error of b.

For the regularization value of $\lambda = 100$, **report the condition number with the standardization technique applied and without**.

Solution:

Condition number (non-standardized): 52711695.28504816

Condition number (standardized): 444.7472601891519

6 Honor Code

1. **List all collaborators. If you worked alone, then you must explicitly state so.**

   Solution: N/A

2. **Declare and sign the following statement**:

   "I certify that all solutions in this document are entirely my own and that I have not looked at anyone else's solution. I have given credit to all external sources I consulted."

   Signature : Zhe Wee Ng (Derrick)

   While discussions are encouraged, *everything* in your solution must be your (and only your) creation. Furthermore, all external material (i.e., *anything* outside lectures and assigned readings, including figures and pictures) should be cited properly. We wish to remind you that the consequences of academic misconduct are *particularly severe*!

   Solution: I certify that all solutions in this document are entirely my own and that I have not looked at anyone else's solution. I have given credit to all external sources I consulted.