DSL501 : Machine Learning - Assignment 1

Name: Ashutosh Kumar Jha

Roll Number: 12340390

Subject Name: Machine Learning

Subject Code: DSL501

Q1. Feature Elimination with Regularization

1.1 Why are ℓ_1 and ℓ_2 regularizations responsible for either feature elimination or feature weight reduction?

Consider a linear model with parameter vector $\beta \in \mathbb{R}^p$. The (penalized) empirical risk for squared-error loss is:

$$\mathcal{L}(\beta) \;=\; \frac{1}{2n}\sum_{i=1}^{n}(y_i - x_i^\top \beta)^2 \;+\; \lambda\,\Omega(\beta),$$

where $\Omega(\beta)$ is the penalty:

- ℓ_2 penalty: $\Omega(\beta) = \frac{1}{2}\|\beta\|_2^2 = \frac{1}{2}\sum_j \beta_j^2$.
- ℓ_1 penalty: $\Omega(\beta) = \|\beta\|_1 = \sum_j |\beta_j|$.

Intuition:

- ℓ_2 penalty penalizes large squared weights. The effect is to *shrink* coefficients continuously toward zero; solutions remain dense (rarely exactly zero) because the penalty is differentiable and imposes a quadratic cost for deviations from zero.
- ℓ_1 penalty imposes a constant (linear) cost per unit magnitude and is not differentiable at 0. This non-differentiability allows the optimum to occur exactly at zero for some coordinates, resulting in *sparse* solutions (feature elimination).

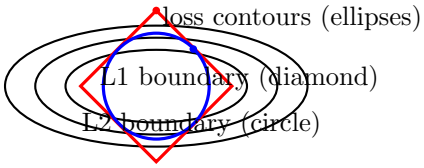
1.2 Explain this using appropriate curves/contours to show how the penalty terms interact with the loss surface.

We show the typical geometric view in the 2D parameter case (β_1, β_2). The unregularized squared-error loss has elliptical contours (level sets). The regularizer constrains the solution to lie inside a small region:

$$\text{LASSO: } \|\beta\|_1 \le t \quad \text{(a diamond)}; \qquad \text{Ridge: } \|\beta\|_2^2 \le t \quad \text{(a circle)}.$$

The penalized solution can be seen as the first point where the elliptical loss contour touches the constraint region. For the ℓ_1 diamond, touching often occurs at an axis (a corner), producing $\beta_j = 0$. For ℓ_2 (a round ball), touching occurs on a smooth boundary seldom exactly on an axis, producing small but nonzero coefficients.

TikZ illustration: (diamond vs circle vs elliptical contours)



1.3 Specifically, illustrate and explain the curve of Elastic Net and how it combines the properties of ℓ_1 and ℓ_2.

Elastic Net penalty mixes both:

$$\Omega_{\text{EN}}(\beta) \;=\; \alpha\|\beta\|_1 \;+\; \frac{1-\alpha}{2}\|\beta\|_2^2, \qquad \alpha \in [0,1].$$

Equivalent penalized objective:

$$\mathcal{L}_{\text{EN}}(\beta) \;=\; \frac{1}{2n}\sum_{i=1}^{n}(y_i - x_i^\top\beta)^2 + \lambda\Big(\alpha\|\beta\|_1 + \tfrac{1-\alpha}{2}\|\beta\|_2^2\Big).$$

Properties:

- When $\alpha = 1$: pure LASSO (ℓ_1) — strong sparsity.
- When $\alpha = 0$: pure Ridge (ℓ_2) — shrinkage only.
- Intermediate α yields a *rounded* diamond: the ℓ_1 diamond corners are smoothed by the ℓ_2 term. This tends to keep some sparsity (zeros possible) while stabilizing coefficients (grouping effect and better numerical conditioning).

Elastic net geometry: the feasible set is the Minkowski sum (informally) of a diamond and a circle producing a rounded-diamond region; intersections with elliptical loss contours can still occur near coordinate axes (so sparsity remains possible) but with less aggressive bias than pure LASSO.

1.4 Mathematical reason for sparsity of ℓ_1

Consider one-dimensional coordinate-wise soft-thresholding solution for orthonormal design. For simplicity: OLS estimate $\hat{\beta}_j^{\text{OLS}}$. Under LASSO (with one coordinate at a time), closed-form is:

$$\hat{\beta}_j^{\text{LASSO}} = \text{sign}(\hat{\beta}_j^{\text{OLS}}) \cdot \max\left\{|\hat{\beta}_j^{\text{OLS}}| - \lambda,\, 0\right\}.$$

Thus when $|\hat{\beta}_j^{\text{OLS}}| \le \lambda$ the coefficient is exactly zero. The ℓ_2 solution (Ridge) instead:

$$\hat{\beta}_j^{\text{Ridge}} = \frac{\hat{\beta}_j^{\text{OLS}}}{1+\lambda},$$

which is shrunk but never exactly zero for finite λ.

Conclusion: ℓ_1 promotes exact zeros via non-differentiability at 0; ℓ_2 yields continuous shrinkage. Elastic Net interpolates between these behaviors.

Q2. Choice of Logistic Function in Classification

2.1 Why do we use the logistic (sigmoid) function for classification instead of other functions?

The logistic (sigmoid) function is

$$\sigma(z) = \frac{1}{1+e^{-z}} \in (0,1).$$

Key reasons it is preferred for binary classification (logistic regression):

1. **Probability output:** $\sigma(\cdot)$ maps any real-valued linear predictor $z = x^\top \beta$ into $(0,1)$, making it a natural model for a Bernoulli probability:

$$p(y=1 \mid x) = \sigma(x^\top \beta).$$

2. **Canonical link of Bernoulli in exponential family:** The Bernoulli pmf can be written in exponential family form with natural parameter $\theta = \log \frac{p}{1-p}$ (the log-odds). The logistic link sets the linear predictor equal to the natural parameter:

$$\log \frac{p}{1-p} = x^\top \beta,$$

   which yields analytical simplifications and desirable properties (e.g. convex negative log-likelihood).

3. **Convexity of negative log-likelihood:** For logistic regression, the negative log-likelihood (over data) is convex in β; this ensures unique global optimum and straightforward optimization with gradient-based methods.

4. **Simple derivative:** $\sigma'(z) = \sigma(z)\big(1-\sigma(z)\big)$ which simplifies gradient/Hessian computations in optimization and yields stable numerical behavior.

5. **Interpretability:** The coefficients β have odds-ratio interpretation: increasing x_j by 1 multiplies the odds by $\exp(\beta_j)$.

6. **Asymptotic/statistical convenience:** Because logistic regression is a generalized linear model (GLM) with canonical link, asymptotic theory (MLE consistency, asymptotic normality, standard errors via Fisher information) is straightforward.

2.2 In particular, why does the logistic function, being part of the exponential function family, make it mathematically suitable for classification?

The Bernoulli (binary) distribution is a one-parameter exponential family:

$$p(y;\theta) = \exp\big(y\theta - A(\theta)\big)h(y),$$

with natural parameter $\theta = \log \frac{p}{1-p}$ and log-partition $A(\theta) = \log(1+e^\theta)$. Choosing the logit as the link function (i.e., $\theta = x^\top \beta$) makes logistic regression the canonical GLM for binary outcomes. This offers:

- **Concise likelihood form:** The Bernoulli log-likelihood for samples (x_i, y_i) is

$$\ell(\beta) = \sum_{i=1}^{n} \big(y_i x_i^\top \beta - \log(1+e^{x_i^\top \beta})\big).$$

- **Convenient derivatives:**

$$\nabla_\beta \ell(\beta) = \sum_{i=1}^{n} x_i \big(y_i - \sigma(x_i^\top \beta)\big).$$

- **Convexity:** $-\ell(\beta)$ is convex, enabling stable optimization and global convergence for methods such as (regularized) Newton or stochastic gradient descent.

Alternative link choices: Other functions could map real numbers to $(0, 1)$ (e.g., probit with $\Phi(z)$, Cauchy-based links). But logistic is computationally convenient, has a simple closed form, and directly corresponds to the canonical parameter of the Bernoulli exponential family.

Q3. Likelihood Under IID Assumption

The classical logistic regression likelihood assumes IID samples (x_i, y_i), $i = 1, \ldots, n$, so:

$$p(\mathbf{y} \mid X, \beta) \;=\; \prod_{i=1}^{n} p(y_i \mid x_i, \beta) \;=\; \prod_{i=1}^{n} \sigma(x_i^\top \beta)^{y_i} \big(1 - \sigma(x_i^\top \beta)\big)^{1-y_i},$$

and the log-likelihood is the sum of per-sample log-likelihoods:

$$\ell_{\text{IID}}(\beta) \;=\; \sum_{i=1}^{n} \Big[y_i \log \sigma(x_i^\top \beta) + (1 - y_i) \log \big(1 - \sigma(x_i^\top \beta)\big) \Big].$$

However, when the IID assumption is *not* valid, the joint distribution does not factorize into independent marginal conditionals. A correct approach is to start from the chain rule (general factorization):

$$p(y_1, \ldots, y_n \mid X, \Theta) = \prod_{i=1}^{n} p\big(y_i \mid y_{1:i-1}, X, \Theta\big),$$

where Θ denotes model parameters (e.g., β and possibly extra dependency parameters).

3.1 General non-IID formulation (chain rule)

Write the likelihood as:

$$\mathcal{L}(\Theta) \;=\; p(\mathbf{y} \mid X, \Theta) = \prod_{i=1}^{n} p(y_i \mid y_{1:i-1}, X, \Theta).$$

Take the log:

$$\ell(\Theta) = \sum_{i=1}^{n} \log p(y_i \mid y_{1:i-1}, X, \Theta).$$

To proceed we must model the conditional distributions $p(y_i \mid y_{1:i-1}, X, \Theta)$. Different dependency structures yield different likelihoods. Below we present a concrete and commonly used non-IID model: a *first-order Markov logistic model* (autoregressive logistic).

3.2 Example: Markov (autoregressive) logistic model

Assume responses follow a first-order Markov dependence: y_i depends on x_i and the previous response y_{i-1}. Parametrize:

$$p\big(y_i = 1 \mid x_i, y_{i-1}, \beta, \gamma\big) = \sigma\big(x_i^\top \beta + \gamma\, y_{i-1}\big),$$

where $\gamma \in \mathbb{R}$ captures serial dependence. (For $i = 1$ we may assume a given initial distribution $p(y_1 \mid x_1)$ or treat y_0 as known/0.)

Then the joint likelihood becomes:

$$\mathcal{L}(\beta, \gamma) \;=\; \prod_{i=1}^{n} \sigma\big(x_i^\top \beta + \gamma y_{i-1}\big)^{y_i} \big(1 - \sigma(x_i^\top \beta + \gamma y_{i-1})\big)^{1-y_i}.$$

Log-likelihood:

$$\ell(\beta, \gamma) \;=\; \sum_{i=1}^{n} \Big[y_i \log \sigma(x_i^\top \beta + \gamma y_{i-1}) + (1 - y_i) \log \big(1 - \sigma(x_i^\top \beta + \gamma y_{i-1})\big) \Big].$$

Gradient (score) for optimization: Using $\sigma'(z) = \sigma(z)(1 - \sigma(z))$, the gradient components are:

$$\frac{\partial \ell}{\partial \beta} = \sum_{i=1}^{n} x_i \big(y_i - \sigma(x_i^\top \beta + \gamma y_{i-1})\big), \qquad \frac{\partial \ell}{\partial \gamma} = \sum_{i=1}^{n} y_{i-1} \big(y_i - \sigma(x_i^\top \beta + \gamma y_{i-1})\big).$$

These plug into gradient-based optimization (e.g., numerical maximization of ℓ).

3.3 Other non-IID structures

Grouped/clustered data: Suppose data are grouped into G clusters and within a cluster observations are dependent. Let cluster g have observations indexed by $i \in C_g$. With a random-effect (mixed) logistic model:

$$p(y_{gi} = 1 \mid x_{gi}, b_g) = \sigma(x_{gi}^\top \beta + b_g),$$

where $b_g \sim \mathcal{N}(0, \tau^2)$ are cluster-specific random effects. The marginal likelihood for cluster g is:

$$p(\mathbf{y}_g \mid X_g, \beta, \tau^2) = \int \Big[\prod_{i \in C_g} p(y_{gi} \mid x_{gi}, b_g, \beta) \Big] \, p(b_g) \, db_g,$$

and the overall likelihood is the product over clusters:

$$\mathcal{L}(\beta, \tau^2) = \prod_{g=1}^{G} p(\mathbf{y}_g \mid X_g, \beta, \tau^2).$$

This is not an IID product over individual samples; evaluation requires integration over random effects (approximation via Laplace, Gauss-Hermite quadrature, or hierarchical Bayesian sampling).

Arbitrary dependence: In the most general case only the chain-rule factorization is guaranteed:

$$p(\mathbf{y} \mid X, \Theta) = \prod_{i=1}^{n} p(y_i \mid y_{1:i-1}, X, \Theta).$$

Specific modeling assumptions are needed to make this tractable: Markov assumptions, conditional independence given latent variables, copula-based dependencies, etc.

3.4 Summary / Key points

- IID assumption leads to product-form likelihood and a sum-form log-likelihood; optimization is straightforward and scales linearly in n.
- Non-IID data require modeling dependencies explicitly. The chain rule yields the exact factorization, but to obtain a practical likelihood we must (a) assume a Markov/conditional structure or (b) introduce latent variables/random effects and marginalize them out.
- Examples provided: Markov-logistic (autoregressive) model (simple and yields a tractable conditional likelihood) and mixed-effects logistic (clustered dependence) that leads to marginal likelihoods requiring integration over random effects.