A1

(a)

L1 regularization results in sparsity because it applies a constant shrinkage force to all weights, regardless of their size. When a weight is small enough, this force pushes it past zero.

L2 regularization, in contrast, applies a proportional shrinkage force, meaning the smaller a weight is, the smaller the force acting on it. As a result, weights gradually approach zero but never become exactly zero, leading to small but nonzero values instead of sparsity.

(b)

A potential upside of the given regularizer is that it penalizes large weights less than L1 or L2 regularization because the sum $\sum_i |w_i|^{0.5}$ grows more slowly than $\sum_i |w_i|$ or $\sum_i w_i^2$, meaning large weights contribute less to the total penalty. This allows important features to remain nonzero while still enforcing sparsity.

A downside is that it may lead to over-sparsification because it disproportionately penalizes smaller weights, potentially eliminating moderately useful features that would have been retained under L1 or L2 regularization, which can harm model performance.

(c)

True. If the step size in gradient descent is too large, the updates may overshoot the optimal solution, causing the loss function to oscillate or even diverge instead of converging. This prevents the algorithm from settling at a minimum, making training unstable and ineffective.

(d)

An advantage of SGD over Batch Gradient Descent is that it updates the model parameters after each data point, making it significantly faster and more scalable for large datasets. A disadvantage of SGD is that its updates have higher variance, causing the optimization process to oscillate and making convergence less stable compared to Batch Gradient Descent.

(e)

Gradient Descent is necessary for logistic regression because the loss function (log-loss) is non-linear and does not have a closed-form solution due to the presence of the sigmoid function, which introduces exponentials into the optimization equation. In contrast, linear regression has a closed-form solution, allowing the optimal weights to be computed directly without iterative optimization.