REVIEW QUESTIONS

1. What is Data Augmentation?

It is the technique to create artificial data from training data.

Its used for-

**Improves Generalization:** Real-world data comes in many variations. By training on augmented data with slight alterations (like flipped images or brightness changes), the neural network learns to recognize the underlying features regardless of these variations. This helps it perform well on new, unseen data.

**Reduces Overfitting:** Neural networks can memorize the training data too well, hindering their ability to adapt to new data (overfitting). Data augmentation combats this by introducing variations, making it harder for the network to simply memorize and forcing it to learn the general patterns.

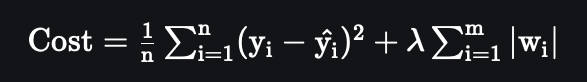
2.What is L1 and L2 also state differences

between them.

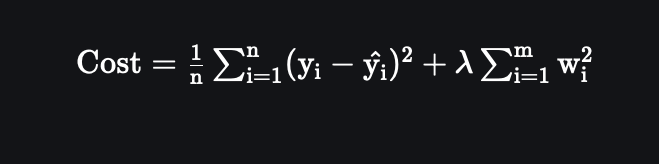
**1.Lasso Regression**

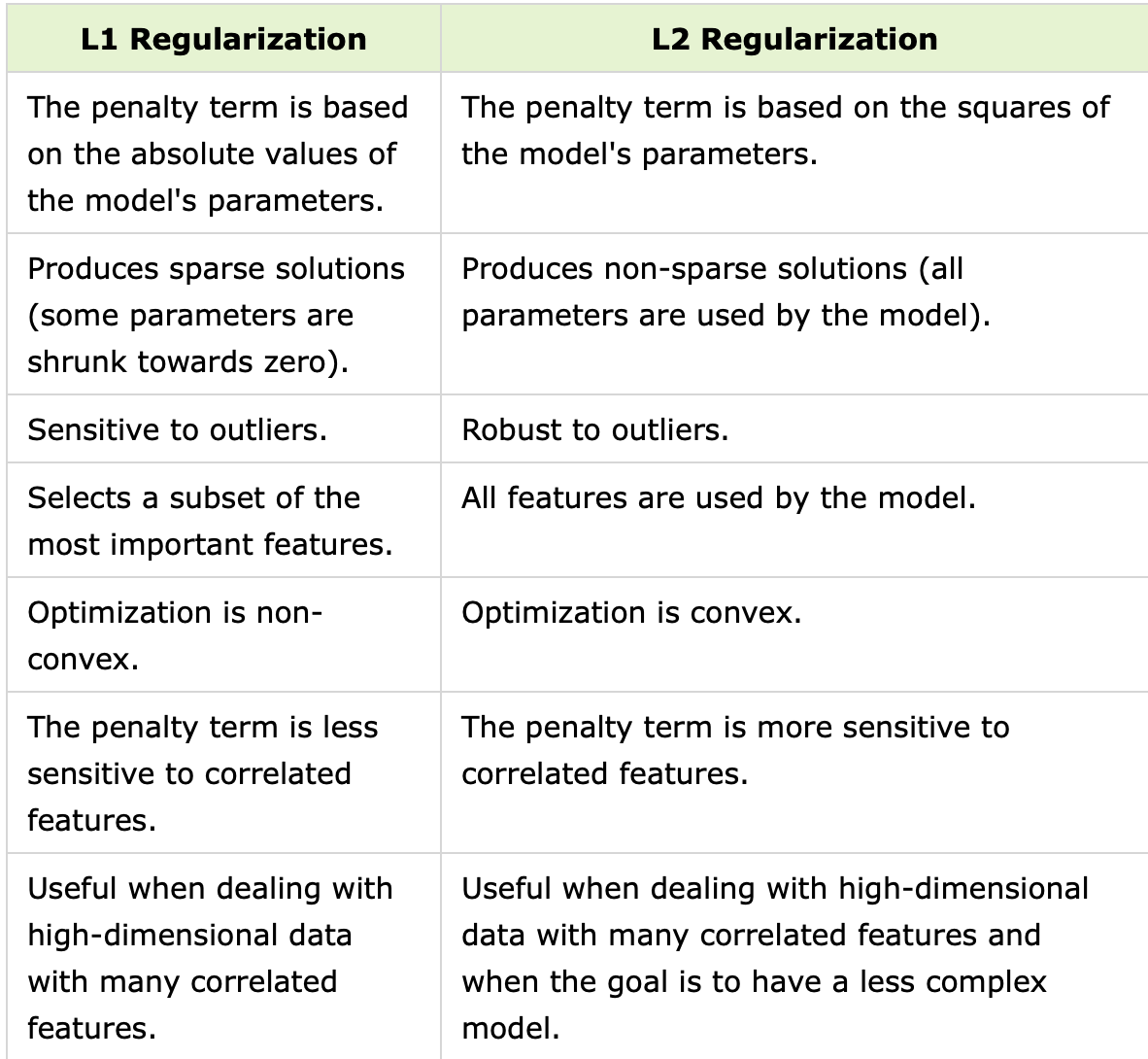
LASSO (Least Absolute Shrinkage and Selection Operator)

regression is a type of regression model that employs L1

Regularization. It adds the absolute magnitude of coefficients as a penalty term to the loss function, aiding in feature selection by pushing insignificant feature weights close to zero.

**2.Ridge Regression**

A regression model that uses the L2 regularization technique is called Ridge regression. Ridge regression adds the “squared magnitude” of the coefficient as a penalty term to the loss function(L).

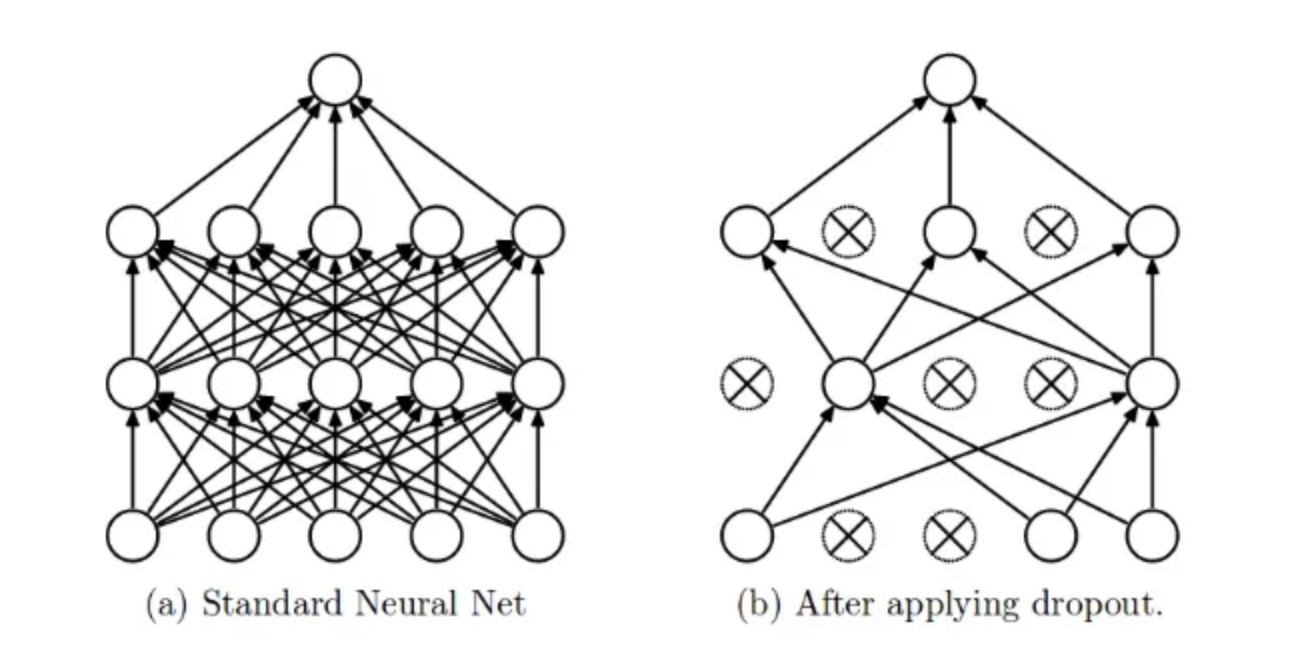


3.What is dropout?

The term “dropout” refers to dropping out the nodes (input and hidden layer) in a neural network . All the forward and backwards connections with a dropped node are temporarily removed, thus creating a new network architecture out of the parent network. The nodes are dropped by a dropout probability of p.

Dropout combats overfitting by solving two key issues:

**Co-adapted Neurons:**

Problem: Neurons become overly reliant on specific activation patterns from others, making them vulnerable to overfitting.

Dropout Solution: Randomly dropping neurons forces them to learn independently and collaborate differently on each training pass, preventing overreliance on specific co-activations.

**Overly Influential Features:**

Problem: Training can lead to overemphasis on specific features, harming generalization.

Dropout Solution: By dropping neurons, the network can't solely focus on those features. It must learn robust features consistent across the data.

4.Choosing the correct gradient.

**1. Batch Gradient Descent:**

* **Advantages:**
  + Simple and easy to implement.
  + Can be computationally efficient for small datasets.
  + Produces a stable error gradient, leading to smoother convergence (sometimes).
* **Disadvantages:**
  + Slow for large datasets as it requires calculating
  + gradients for all data points before updating the model.
  + May get stuck in local minima due to the stable
  + gradient.
  + Requires the entire dataset to be in memory, which can be impractical for large datasets.

**2. Stochastic Gradient Descent (SGD):**

* **Advantages:**
  + Faster than batch gradient descent for large datasets as it updates the model after each data point.
  + Less prone to getting stuck in local minima due to the stochastic (random) nature of updates.
  + Requires less memory as it only processes one data point at a time.
* **Disadvantages:**
  + Updates can be noisy, leading to more erratic convergence compared to batch GD.
  + May require more tuning of the learning rate to achieve optimal convergence.

**3. Mini-batch Gradient Descent:**

* **Advantages:**
  + Offers a balance between batch GD and SGD.
  + Processes data in small batches, reducing memory usage compared to batch GD.
  + Can be faster than batch GD for large datasets while providing smoother convergence than SGD.
* **Disadvantages:**
  + Still requires some tuning of the batch size for optimal performance.
  + Not as simple to implement as batch GD or SGD.

**Choosing the Right Variant:**

The best choice depends on your specific dataset size and

characteristics. Here's a general guideline:

* **Small datasets:** Batch GD can be a good option due to its simplicity.
* **Large datasets:** Mini-batch GD often offers a good balance between speed and convergence.
* **Very large datasets:** SGD might be necessary due to
* memory limitations, but may require more experimentation with the learning rate.

5.What is EMWA?

The Exponentially Weighted Moving Average (EWMA) is

commonly used as a smoothing technique in time series.

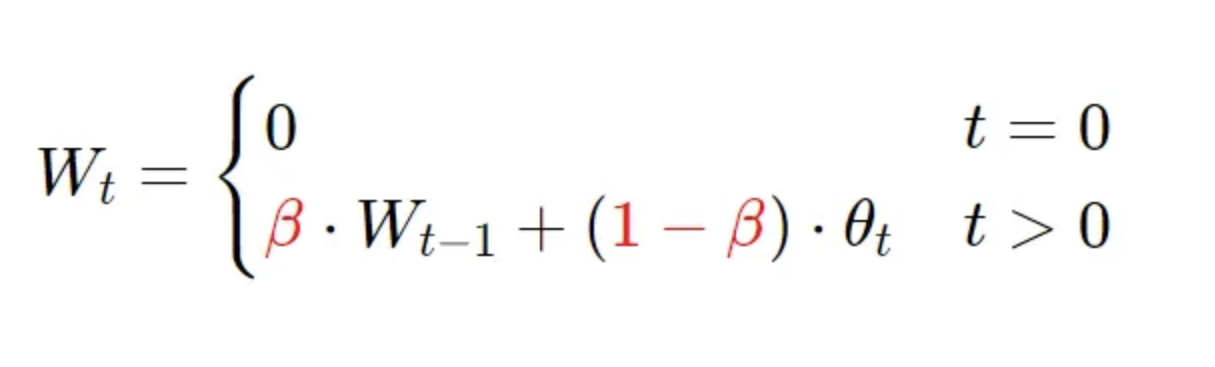
However, due to several computational advantages

(fast, low-memory cost), the EWMA is behind the scenes of many optimization algorithms in deep learning, including Gradient

Descent with Momentum,

RMSprop, Adam, etc.

In order to compute the EWMA, you must define one parameter β



Above is the formula for computation of EMWA,where

β= weight parameter

Wt=EWA for the day t

theta(t)=temperature of day t

If -

β is high-more wight given to past observations, making EMWA respond slowly to changes in the data, this results in smooth average.

β is low-more weight given to recent observation, making EMWA more responsible to changes in data, this results in smooth average that adapts to new data.

5.What is bias correction?

Bias correction is a technique used to address systematic errors in data, particularly in fields that rely on computer models.

Data Discrepancies: Imagine a weather forecasting model that predicts temperatures a few degrees higher than what's actually observed. This difference is a bias.

Correcting the Bias: Bias correction involves statistically adjusting the model's outputs to better match real-world observations. In the weather example, this might involve subtracting a constant value from the predicted temperatures to account for the model's tendency to overestimate.

Applications-

Climate modelling

Machine learning

\*DESIGN A NN

https://github.com/rampofin/mrm/blob/main/Aashirvad\_Research\_AI\_9\_22\_05\_2024.ipynb