**General Linear Model (GLM):**

1. The purpose of the General Linear Model (GLM) is to model the relationship between a dependent variable (response) and one or more independent variables (predictors) while considering the influence of error terms. It is a flexible framework that encompasses various statistical models, including simple and multiple linear regression, analysis of variance (ANOVA), analysis of covariance (ANCOVA), and others.

2. The key assumptions of the General Linear Model include:

a. Linearity: The relationship between the dependent variable and the predictors is assumed to be linear.

b. Independence: Observations are assumed to be independent of each other.

c. Homoscedasticity: The variance of the errors is constant across all levels of the predictors.

d. Normality: The errors are assumed to be normally distributed.

3. In a GLM, the coefficients represent the change in the dependent variable associated with a one-unit change in the corresponding predictor, while holding other predictors constant. The coefficients indicate the direction and magnitude of the relationship between the predictors and the response variable.

4. Univariate GLM involves a single dependent variable and one or more independent variables, whereas multivariate GLM deals with multiple dependent variables simultaneously. Univariate GLM is suitable when analyzing the effect of predictors on a single outcome, while multivariate GLM is used when studying the relationships between predictors and multiple outcomes.

5. Interaction effects in a GLM occur when the effect of one predictor on the dependent variable depends on the level of another predictor. In other words, the relationship between one predictor and the dependent variable is not constant across different levels of another predictor. Interaction terms allow the model to account for this variation and capture more complex relationships between the variables.

6. Categorical predictors are typically handled by converting them into dummy variables (also known as one-hot encoding). Each category of the categorical variable is represented by a binary dummy variable, which takes a value of 0 or 1, indicating the absence or presence of that category.

7. The design matrix in a GLM is a matrix that represents the relationship between the dependent variable and the independent variables. Each row of the design matrix corresponds to an observation, while each column corresponds to a predictor. The design matrix is used to estimate the model parameters and fit the GLM to the data.

8. The significance of predictors in a GLM is typically tested using hypothesis tests, such as t-tests or F-tests, depending on the specific type of GLM and the structure of the model. The p-values associated with these tests indicate whether the predictors have a statistically significant effect on the dependent variable.

9. Type I, Type II, and Type III sums of squares are different methods of partitioning the variability in the dependent variable explained by the predictors. They are used in ANOVA to assess the significance of individual predictors or groups of predictors. The choice of sum of squares depends on the research question and the specific hypotheses being tested.

10. Deviance in a GLM refers to a measure of the lack of fit of the model to the data. It is analogous to the concept of residuals in linear regression. Deviance is used in hypothesis testing and model comparison, particularly in generalized linear models, where the likelihood ratio test is employed to compare nested models.

**Regression:**

11. Regression analysis is a statistical method used to model the relationship between a dependent variable and one or more independent variables. Its purpose is to understand how the changes in the independent variables are associated with changes in the dependent variable.

12. Simple linear regression involves one dependent variable and one independent variable, whereas multiple linear regression deals with one dependent variable and two or more independent variables. In simple linear regression, the relationship is modeled as a straight line, while in multiple linear regression, the relationship is modeled as a hyperplane in higher dimensions.

13. R-squared (R²) is a measure of the proportion of the variance in the dependent variable that is explained by the independent variables in the regression model. It ranges from 0 to 1, where 0 indicates that the model does not explain any variability, and 1 indicates a perfect fit.

14. Correlation measures the strength and direction of the linear relationship between two variables, while regression aims to model and predict the dependent variable using one or more independent variables.

15. Coefficients in regression represent the estimated changes in the dependent variable associated with one-unit changes in the corresponding independent variable, assuming all other variables remain constant. The intercept is the value of the dependent variable when all independent variables are set to zero.

16. Outliers in regression analysis can distort the model's fit and affect the estimated coefficients. Handling outliers may involve identifying and removing them if they are data errors, transforming the data, or using robust regression techniques that are less sensitive to outliers.

17. Ridge regression is a regularized version of linear regression that adds a penalty term to the loss function to prevent overfitting. It uses L2 regularization to shrink the coefficients towards zero. Ordinary least squares (OLS) regression, on the other hand, does not include any regularization and can be prone to overfitting when the number of predictors is large.

18. Heteroscedasticity in regression occurs when the variance of the residuals is not constant across all levels of the predictors. It can lead to inefficient coefficient estimates and incorrect inference. To address heteroscedasticity, one can use robust standard errors or transform the dependent variable to achieve constant variance.

19. Multicollinearity occurs when two or more independent variables in a regression model are highly correlated, making it challenging to separate their individual effects on the dependent variable. To handle multicollinearity, one can consider removing one of the correlated variables, perform dimensionality reduction, or use regularization techniques.

20. Polynomial regression is a form of regression where the relationship between the dependent variable and the independent variable(s) is modeled as a polynomial function of a certain degree. It is used when the data shows a nonlinear relationship, and higher-order polynomial terms are included to capture the curvature in the data.

**Loss Function:**

21. A loss function measures the discrepancy between the predicted values of a model and the actual target values. Its purpose in machine learning is to guide the optimization process during model training, as the goal is to minimize the loss function to improve the model's performance.

22. A convex loss function has a single global minimum, making optimization easier since any local minimum is also the global minimum. In contrast, a non-convex loss function has multiple local minima, making optimization more challenging.

23. Mean Squared Error (MSE) is a common loss function used for regression problems. It calculates the average of the squared differences between the predicted values and the actual target values. MSE penalizes large errors more heavily than small errors.

24. Mean Absolute Error (MAE) is another loss function used for regression tasks. It calculates the average of the absolute differences between the predicted values and the actual target values. MAE is less sensitive to outliers compared to MSE.

25. Log Loss, also known as cross-entropy loss, is typically used for classification problems in machine learning. It measures the performance of a probabilistic classifier by comparing the predicted probabilities to the actual class labels.

26. The choice of an appropriate loss function depends on the nature of the problem being addressed. For regression tasks, MSE is commonly used when the data contains outliers, while MAE is preferred when robustness to outliers is crucial. For classification problems, log loss is suitable for probabilistic predictions.

**Optimizer (GD):**

31. An optimizer is an algorithm or method used in machine learning to update the parameters of a model during the training process. Its purpose is to minimize the loss function by finding the optimal set of model parameters that best fit the data. Optimization is a critical component of the training process, as it determines how the model learns from the data and converges to a solution.

32. Gradient Descent (GD) is an iterative optimization algorithm used to find the minimum of a loss function. It works by iteratively updating the model parameters in the direction of the steepest descent of the loss function. The steepest descent is determined by the negative gradient (slope) of the loss function with respect to the parameters. By taking small steps in the direction of the negative gradient, the algorithm aims to reach the minimum of the loss function.

33. There are different variations of Gradient Descent, including:

a. Batch Gradient Descent: In this method, the algorithm updates the model parameters using the entire training dataset in each iteration. It computes the average gradient over the entire dataset and takes a step in that direction. Batch GD can be computationally expensive, especially for large datasets.

b. Stochastic Gradient Descent (SGD): SGD updates the model parameters using only one randomly chosen training sample in each iteration. It introduces randomness into the parameter updates and can lead to faster convergence, but it can also be noisy and have high variance in its updates.

c. Mini-batch Gradient Descent: This method combines the benefits of both batch GD and SGD. It updates the model parameters using a small batch of randomly chosen training samples in each iteration. Mini-batch GD strikes a balance between computational efficiency and noise reduction.

34. The learning rate in Gradient Descent is a hyperparameter that controls the step size taken in the direction of the negative gradient during each parameter update. Choosing an appropriate learning rate is crucial for the convergence of the optimization process. If the learning rate is too small, the algorithm may converge very slowly. If it is too large, the algorithm may oscillate around the minimum or even diverge.

35. GD can get stuck in local optima in optimization problems, especially if the loss function is non-convex and has multiple local minima. However, this is less of a concern in deep learning and neural networks, as the loss functions used are usually highly non-convex but have many relatively flat regions. GD can still converge to satisfactory solutions due to the complex landscape of the loss function.

36. Stochastic Gradient Descent (SGD) is a variation of GD where, instead of using the entire training dataset to compute the gradient, it randomly selects one training sample at a time to calculate the gradient and update the parameters. This makes each iteration of SGD computationally faster, but the updates can be noisy and introduce variance into the optimization process.

37. The batch size in GD refers to the number of training samples used in each iteration to compute the gradient and update the parameters. In batch GD, the batch size is the entire training dataset. In mini-batch GD, the batch size is typically a small number, such as 32 or 64. The impact of batch size on training is a trade-off between computation speed and the quality of the updates. Larger batch sizes can lead to more stable updates but may be computationally expensive, while smaller batch sizes introduce more noise but can converge faster.

38. Momentum is a technique used in optimization algorithms, including GD, to accelerate convergence and overcome oscillations during training. It introduces a "velocity" term that influences the direction and magnitude of the parameter updates. Momentum helps the optimizer to keep moving in the same direction when the gradients have consistent signs and increases the step size when the gradients change direction, allowing the optimizer to escape shallow local minima.

39. The main difference between batch GD, mini-batch GD, and SGD lies in the number of training samples used in each iteration to update the model parameters. Batch GD uses the entire dataset, mini-batch GD uses a small randomly chosen subset of the data, and SGD uses a single randomly chosen data point. Batch GD can have slower convergence but more stable updates, while SGD can have faster convergence but noisy updates. Mini-batch GD offers a trade-off between the two approaches.

40. The learning rate affects the convergence of GD by controlling the step size of parameter updates. A larger learning rate can result in faster convergence, but it may cause overshooting and instability, leading to divergence. On the other hand, a smaller learning rate may lead to slow convergence or getting stuck in local minima. Choosing the appropriate learning rate often involves experimentation and tuning, and techniques like learning rate schedules and adaptive learning rate methods (e.g., Adam, RMSprop) are used to dynamically adjust the learning rate during training to improve convergence.

**Regularization:**

41. Regularization is a technique to prevent overfitting and improve model generalization by adding penalty terms to the loss function during training.

42. L1 regularization adds the sum of the absolute values of coefficients, promoting sparsity. L2 regularization adds the sum of squared values of coefficients, encouraging a smoother solution.

43. Ridge regression is a linear regression with L2 regularization, controlling model complexity by shrinking coefficients.

44. Elastic Net combines L1 and L2 penalties, allowing feature selection and handling correlated features.

45. Regularization prevents overfitting by penalizing complex models and encouraging simpler solutions.

46. Early stopping stops training when model performance on a validation set degrades, preventing overfitting.

47. Dropout regularization randomly deactivates neurons during training in neural networks to prevent reliance on specific neurons.

48. The regularization parameter is chosen through hyperparameter tuning and cross-validation.

49. Feature selection explicitly selects important features, while regularization controls model complexity mathematically.

50. Regularized models trade-off between increased bias (simplicity) and decreased variance (generalization).

**Decision Tree:**

61. A decision tree is a predictive model that recursively splits data based on feature values to make decisions and predict outcomes.

62. Splits in a decision tree are made by selecting the best feature and threshold that maximize information gain or minimize impurity.

63. Impurity measures like Gini index and entropy quantify the uncertainty or disorder in data and help find optimal splits in decision trees.

64. Information gain measures how much the impurity of data decreases after a split, indicating the usefulness of a feature for decision making.

65. Missing values in decision trees can be handled by ignoring them during split evaluation or imputing them based on other data.

66. Pruning in decision trees involves removing branches that don't contribute significantly to improve the model's performance, reducing overfitting.

67. A classification tree predicts class labels, while a regression tree predicts numerical values.

68. Decision boundaries in a decision tree are regions that separate data points of different classes based on feature thresholds.

69. Feature importance in decision trees indicates how much each feature influences the model's predictions.

70. Ensemble techniques combine multiple decision trees to create more accurate and robust models, e.g., Bagging, Random Forests, and Boosting.

**Ensemble Techniques:**

71. Ensemble techniques combine multiple models to improve predictive performance and model robustness in machine learning.

72. Bagging (Bootstrap Aggregating) is an ensemble technique that creates multiple models trained on different random subsets of the data and combines their predictions for more accurate results.

73. Bootstrapping in bagging involves sampling the training data with replacement to create multiple subsets, allowing each model to have different data samples.

74. Boosting is an ensemble technique that builds models sequentially, where each new model focuses on the mistakes of the previous models to improve overall accuracy.

75. AdaBoost adjusts the weights of data points during each iteration to prioritize misclassified samples, while Gradient Boosting fits each new model to the residuals of the previous model.

76. Random Forests are an ensemble of decision trees that reduce overfitting and improve generalization by averaging the predictions of multiple trees.

77. Random Forests determine feature importance by measuring how much each feature contributes to reducing impurity in the tree splits.

78. Stacking in ensemble learning combines predictions from multiple base models by training a higher-level model on their outputs.

79. Advantages of ensemble techniques include improved performance, robustness, and handling complex relationships. Disadvantages include increased complexity and computation.

80. The optimal number of models in an ensemble depends on the specific problem and dataset. It is often determined through experimentation and cross-validation.