Absolutely! Here's a clear, concise breakdown of LDA, QDA, and GBM, with an emphasis on how they work and when you might use each:

### 🧠 **LDA — Linear Discriminant Analysis**

#### ➤ What is it?

LDA is a supervised classification technique used when the response variable is categorical. It assumes:

* Each class generates data based on a multivariate normal distribution.
* All classes share the same covariance matrix.

#### ➤ How it works:

* It projects high-dimensional data onto a lower-dimensional space (called discriminant axes) to maximise class separability.
* Then it uses Bayes’ theorem with class-specific normal distributions to classify new observations.

#### ➤ When to use:

* When classes are linearly separable.
* When you believe the covariance structure is similar across classes.

### 🧠 **QDA — Quadratic Discriminant Analysis**

#### ➤ What is it?

QDA is similar to LDA but relaxes the assumption of equal covariance matrices between classes.

#### ➤ How it works:

* Each class gets its own covariance matrix.
* This allows for non-linear decision boundaries.
* Like LDA, it uses Bayes' theorem but with more flexible Gaussian assumptions.

#### ➤ When to use:

* When classes have distinct variance structures or are not linearly separable.
* Needs more data than LDA (more parameters to estimate).

### 🌲 **GBM — Gradient Boosting Machine**

#### ➤ What is it?

GBM is an ensemble method that builds a strong model by combining many weak learners, usually decision trees.

#### ➤ How it works:

* Trees are added sequentially, each one trying to correct the errors made by the previous ones.
* It minimises a loss function (like MSE for regression or log-loss for classification) using gradient descent.

#### ➤ When to use:

* When you want high predictive performance.
* When relationships are non-linear or complex.
* Suitable for both regression and classification.

#### ➤ Downsides:

* Sensitive to hyperparameters (learning rate, number of trees).
* Can overfit if not tuned properly.

### 🧾 Summary Table

| Method | Type | Assumes Normality | Covariance Assumption | Decision Boundary | Notes |
| --- | --- | --- | --- | --- | --- |
| LDA | Classification | ✅ Yes | ✅ Same across classes | Linear | Simple, fast |
| QDA | Classification | ✅ Yes | ❌ Different for each class | Quadratic | More flexible |
| GBM | Regression + Classification | ❌ No | ❌ None | Very flexible | High accuracy, slower |

Absolutely! Here's a clear and concise explanation of Linear and Radial SVMs, especially tailored to help you during your exam:

### 🧠 Support Vector Machines (SVMs): A Quick Overview

SVMs are supervised learning algorithms used for classification (and sometimes regression).  
They work by finding a decision boundary (called a hyperplane) that best separates classes in the feature space.

The "best" hyperplane is the one that maximises the margin—i.e. the distance between the closest data points (called support vectors) and the boundary.

## 🔹 Linear SVM

### ➤ What it does:

A Linear SVM assumes that the classes can be separated using a straight line (in 2D) or a flat hyperplane (in higher dimensions).

### ➤ Suitable for:

* Datasets where classes are linearly separable (or almost separable)
* Simpler problems with less complexity

### ➤ Equation of the decision boundary:

The model tries to find weights w and bias b such that:

f(x)=wTx+b=0

is the separating hyperplane.

### ➤ Example intuition:

Imagine drawing a straight line between two groups of dots on a 2D plane — that’s what a linear SVM does.

## 🔸 Radial SVM (a.k.a. SVM with RBF kernel)

### ➤ What it does:

A Radial Basis Function (RBF) SVM allows for non-linear decision boundaries.  
It maps input features into a higher-dimensional space using a kernel function and finds a linear separator in that space.

### ➤ Why this helps:

Many real-world problems (like image classification or tumour classification) involve non-linear class boundaries.  
The RBF kernel lets SVM handle complex shapes in data.

### ➤ Kernel function used:

K(x,x′)=exp(−γ∥x−x′∥2)

This measures similarity between points: points close to each other have higher similarity.

* γ (gamma) controls the "spread" of influence of each training example.
* A small γ → smoother boundaries (less flexible),  
  a large γ → more flexible model, may lead to overfitting.

### ➤ Example intuition:

Imagine placing a smooth hill at each training point, and then the classifier uses these hills to draw a curvy boundary around one class and away from the other.

| Feature | Linear SVM | Radial SVM (RBF Kernel) |
| --- | --- | --- |
| Decision Boundary | Straight line / hyperplane | Curved / flexible boundary |
| Handles Non-Linearity | ❌ No | ✅ Yes |
| Speed | 🟢 Fast | 🔵 Slower (more complex) |
| Overfitting Risk | 🔵 Lower | 🔴 Higher (needs tuning) |
| Key Parameter | C (penalty) | C and γ (gamma) |

Absolutely! Let's break it down with a bit of flair to make it memorable:

### 🔹 **Linear SVM (Support Vector Machine)**

Think of this as the "ruler and pencil" version of SVM.

* You're trying to draw a straight line (or hyperplane) that best separates two classes (e.g. benign vs malignant tumours).
* It works great when the data is linearly separable, meaning you could physically draw a straight line between the two groups without errors.
* The SVM finds the line that maximises the margin — the distance from the line to the closest points in each class (these are the "support vectors").

🧠 Good for when the relationship between variables and class is relatively simple and linear.

### 🔸 **Radial Basis Function (RBF) SVM**

Now imagine you're bending that pencil into curves — welcome to nonlinear land.

* This SVM uses a Radial Basis Function kernel to transform the input space into a higher-dimensional one.
* In this transformed space, it’s easier to separate the classes with a curved boundary, like wrapping around clusters of data.
* Think of RBF as casting a stretchy net over your data — it warps to fit around points.

🧠 Powerful when the class boundaries are complex or wiggly — like a spiral or blob, not a straight slice.

### 🧪 Analogy Time!

* Linear SVM is like a laser pointer slicing a cake — precise, straight.
* Radial SVM is like a malleable wire cutter — it flexes around the weird shape of your fancy cake to make a clean slice.

If you want, I can also sketch a simple visual of the difference — just let me know!

2022-23 Exam Q2

Certainly! Here's a neatly formatted version of your answers, along with your follow-up questions:

### (a) **Is this a regression or a classification problem? Justify your answer.** [4]

Answer: This is a regression problem because the response variable Salary is a continuous numeric value. The task involves predicting the salary of baseball players based on various predictor variables.

### (b) **What is the name of the model used to generate** mod1**?** [4]

Answer: The model used to generate mod1 is Elastic Net. Specifically, it is an Elastic Net regression model (alpha = 0.5 specifies equal mixing of Lasso and Ridge regularization).

Justification: The function glmnet(x.train, y.train, alpha = 0.5) fits an Elastic Net model, where alpha controls the mix between Lasso (L1) and Ridge (L2) regularization.

### (c) **What type of cross-validation is applied to model 2? Justify your answer.** [4]

Answer: 10-fold cross-validation is applied to model 2.

Justification: The function cv.glmnet() is used to perform cross-validation automatically. By default, cv.glmnet() performs 10-fold cross-validation to select the optimal value of the regularization parameter lambda.

### (d) **Explain the difference between criteria** crit1 **and** crit2**, in one sentence.** [4]

Answer: crit1 is the mean squared error (MSE) on the training set using the model mod1 (Elastic Net with default lambda), while crit2 is the MSE on the training set using mod2, which is the Elastic Net model with the optimal lambda selected by cross-validation.

### (e) **Explain the difference between criteria** crit3 **and** crit4**, in one sentence.** [4]

Answer: crit3 is the mean squared error (MSE) on the test set using mod1 (Elastic Net with default lambda), while crit4 is the MSE on the test set using mod2, which is the Elastic Net model with the optimal lambda selected by cross-validation.

### (f) **Comment on the difference between the medians of values contained in** crit2 **and** crit4**, in two sentences maximum.** [5]

Answer: The median of values in crit2 is typically higher than the median of values in crit4, indicating that the model selected using cross-validation (mod2) generalizes better to the test data than the model with the default lambda (mod1). This suggests that using cross-validation to select the best lambda improves the model's predictive performance on unseen data.

### (g) **Explain the difference between the variabilities of values contained in** crit2 **and** crit4**, in two sentences maximum.** [5]

Answer: The variability (spread) of values in crit4 is typically lower than in crit2, indicating that the model with the optimal lambda selected by cross-validation (mod2) provides more stable predictions across different training-test splits. This suggests that tuning the regularization parameter reduces overfitting and improves the model’s consistency on the test set.