# Patient Risk Prediction – Healthcare Project Viva Questions & Answers

## 🎓 General Project Understanding

### What is the objective of your project?

To predict the risk level of patients based on their age and other attributes using machine learning, helping hospitals prioritize care and allocate resources efficiently.

### Why did you choose logistic regression for risk prediction?

Logistic regression is effective for binary and multi-class classification problems. It provides a simple and interpretable model for predicting categorical outcomes like risk levels.

### What kind of data did you use, and how was it collected or generated?

The data includes patient demographics and hospital visit details. It was simulated and structured in CSV format for training and testing the model.

### Explain the schema design (fact and dimension tables) and why you used a star schema.

The schema includes a central fact table for patient risk and dimension tables for patient, hospital, and diagnosis details. A star schema simplifies queries and improves performance for analytical tasks.

## 🧠 Technical Questions

### Why did you use Spark for this project?

Spark enables distributed data processing and is ideal for handling large datasets efficiently. It integrates well with machine learning libraries like MLlib.

### What is MLlib and how does it support machine learning in Spark?

MLlib is Spark’s scalable machine learning library. It provides tools for classification, regression, clustering, and more, using Spark’s distributed architecture.

### How does Spark handle large datasets efficiently?

Spark uses in-memory computation and distributed processing across clusters, which speeds up data transformations and model training.

### What is a VectorAssembler and why is it used?

VectorAssembler combines multiple feature columns into a single vector column required by Spark ML algorithms.

## Modeling

### What are the input features and target variable in your model?

The input feature is 'age', and the target variable is 'risk\_label', which indicates the patient's risk level.

### How did you split the data into training and testing sets?

Using Spark’s randomSplit() method with a 70:30 ratio to ensure proper model training and evaluation.

### What metrics did you use to evaluate your model?

BinaryClassificationEvaluator was used to calculate accuracy based on predicted and actual labels.

### What does the accuracy score tell you about your model’s performance?

It indicates how well the model predicts the correct risk labels. Higher accuracy means better performance.

## 📊 Data Analysis & Visualization

### What does the prediction distribution chart show?

It shows the number of patients predicted to be in each risk category, helping visualize model output and detect bias.

### How balanced are the predicted risk labels?

The chart helps assess label balance. If one label dominates, it may indicate class imbalance.

### Did you face any class imbalance issues? If yes, how did you handle them?

The dataset was small and balanced. For larger datasets, techniques like resampling or class weighting can be used.

## 🛠️ Improvement & Deployment

### How would you improve the model’s accuracy?

By adding more features like source, hospital ID, and diagnosis, and using feature encoding for categorical variables.

### What other features could be added to enhance prediction?

Features like patient history, diagnosis type, and visit frequency could improve prediction accuracy.

### Can this model be deployed in a real hospital system? How?

Yes, by integrating it with hospital databases and using real-time data pipelines for prediction and triage.

### How would you handle real-time data in Spark?

Using Spark Streaming to process and analyze data in real-time from sources like Kafka or REST APIs.

## 📁 File & Notebook Management

### How did you manage your data files and notebooks in Databricks?

CSV files were uploaded to DBFS, and notebooks were used for code execution, model training, and visualization.

### How did you save and share your predictions?

Predictions were saved as CSV files in DBFS and can be downloaded or shared via Databricks workspace.

# Visual Diagrams and Charts

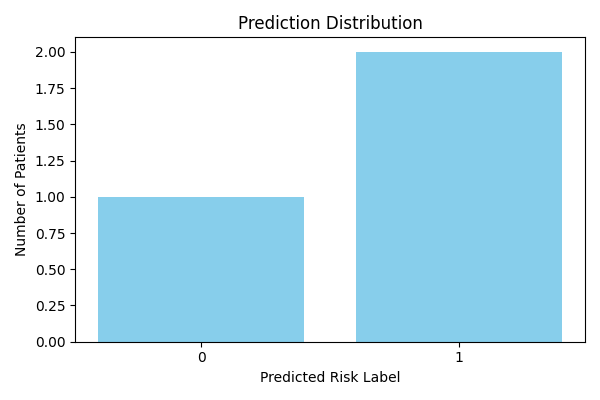
## Star Schema Design

The star schema consists of a central fact table 'fact\_patient\_risk' and three dimension tables: 'dim\_patient', 'dim\_hospital', and 'dim\_diagnosis'. This design simplifies analytical queries and improves performance.

## Model Workflow

The workflow includes: Data Loading → Feature Engineering → Train-Test Split → Model Training → Prediction → Evaluation → Visualization.

## Prediction Distribution Chart



This chart shows the number of patients predicted to be in each risk category by the logistic regression model.