Summarize the Key Points: What are the main objectives of the chapter?

The chapter starts with discussing why it is so difficult to build a model for any classification problem statement. The main objective of the chapter is to guide the reader in choosing the right classification model for their machine learning task. Finally it highlights the importance of understanding the strengths and weaknesses of different classification models and emphasizes the fact that the best model depends on the specific scenario and business case.

Challenges in Model Selection: Why is selecting the right classification model often difficult? Discuss the factors that contribute to this complexity.

Selecting the right classification model is often difficult because each model has its own strengths and weaknesses in different situations. There is no universal flowchart to determine the best model. Infact, there are several factors that play a role in selecting the correct model for instance how much data do you have, the type of data at hand, the desired outcome, visualization needs, and resource limitations. All these factors contribute to the complexity of model selection.

Write a summary of the strengths and weaknesses of all the ML models mentioned in the chapter.

● Logistic Regression:

- **Strengths**: The simplicity of the algorithm and the ease of interpretation of the results is the biggest strength of logistic regression.
- **Weaknesses**: Since this algorithm is limited to binary classification problems, which is less seen in real life scenarios, it has limited usability. For multiclass classification you have to use the softmax version of this!!

● Naive Bayes:

- **Strengths**: It usually gives High bias/low variance performance and works well with limited data. It has exponentially fast training and is good for continuous new data.
- **Weaknesses**: May not perform well with large and complex datasets since this simple Bayes algorithm based method is not suitable for complex hypotheses.

● k-Nearest Neighbor:

- **Strengths**: It is simple to understand and implement, it has no training time involved and hence is less compute heavy compared to other algorithms.

- **Weaknesses**: Feature engineering can dramatically impact the results of KNN and hence it can be easily fooled by irrelevant attributes in the dataset. Since it does not have training time, it does all the computation during testing and hence it has a huge testing time compared to other algorithms. Due to this, if the dataset is huge, it also requires huge cache memory to store the entire dataset in memory during test time.

- **Decision Trees:**
  - **Strengths**: It is relatively fast compared to other algorithms and is easy to understand and visualize.
  - **Weaknesses**: Decision Trees are more prone to overfitting compared to other algorithms and hence regularization techniques like tree pruning are necessary to implement while using this algorithm.

- **Support Vector Machine:**
  - **Strengths**: They are extremely accurate and resistant to overfitting. It also comes with the advantage of handling non linear classification and it does it in a fraction of training / compute time.
  - **Weaknesses**: This algorithm slows down with increase in no of classes.

- **Neural Networks:**
  - **Strengths**: Excellent for modeling nonlinear data with many input features which is usually the scenario in real life situations and hence this algorithm can solve complex problems.
  - **Weaknesses**: It is computationally extremely expensive and difficult to interpret the model. Fine-tuning the model also has its own challenges in different fields of application of this algorithm!!