DATA ANALYTICS AND COGNOS:

Project: Customer Churn Production

Phase -2

Advanced machine learning techniques :

1. Deep Learning:
This involves neural networks with many layers, capable of handling complex tasks like image and speech recognition.

2. Reinforcement Learning:
Algorithms learn by interacting with an environment, often used in gaming, robotics, and autonomous systems.

3. Generative Adversarial Networks (GANs):
Used for generating data that resembles real data, commonly employed in image generation and data augmentation.

4. Transfer Learning:
Leveraging pre-trained models on new, related tasks, which can significantly reduce training time and data requirements.

5. Natural Language Processing (NLP):
Techniques for understanding and generating human language, used in chatbots, translation, and sentiment analysis.

6. Ensemble Learning:
Combining multiple models to improve overall performance, often seen in random forests, gradient boosting, and stacking.

7. AutoML:
Automated machine learning tools and platforms that streamline the model development process, making it more accessible to non-experts.

8. Explainable AI (XAI):
Techniques that provide insights into how AI models make decisions, crucial for transparency and trust.

9. Time Series Analysis:
Specialized methods for handling data with temporal dependencies, used in forecasting and trend analysis.

10. Anomaly Detection:
Identifying unusual patterns or outliers in data, important for fraud detection and system monitoring

Ensemble models

1. Bagging (Bootstrap Aggregating):
Involves training multiple instances of the same base model on different subsets of the training data and then averaging or voting on their predictions. Random Forest is a common example of a bagging ensemble.

2. Boosting:
Builds a sequence of base models where each subsequent model focuses on correcting the errors of the previous ones. AdaBoost and Gradient Boosting are widely used boosting algorithms.

3. Stacking:
Combines the predictions of multiple base models by training a meta-model that learns how to best weigh or combine their outputs. It involves a higher-level model that takes the predictions of the base models as inputs

4. Random Subspace Method:
Similar to bagging but instead of bootstrapping the data, it randomly selects a subset of features for each base model. This is often used with high-dimensional data

Feature Engineering

1. Feature Scaling:
Normalize or standardize numerical features to ensure they have a consistent scale. This is important for algorithms sensitive to feature scales, like SVM or K-Means.

2. One-Hot Encoding:
Convert categorical variables into binary vectors to make them suitable for machine learning algorithms that require numerical input.

3. Feature Extraction:
If you have high-dimensional data, consider techniques like Principal Component Analysis (PCA) or t-Distributed Stochastic Neighbor Embedding (t-SNE) to reduce dimensionality while preserving important information.

4. Binning or Discretization:
Convert continuous variables into discrete bins to capture non-linear relationships and reduce noise in the data.

5. Feature Crosses:
Create new features by combining two or more existing features. This can help capture interactions between variables.

6. Time-Based Features:
If working with time-series data, generate features like lag values, moving averages, or time-based aggregations.

7. Text Feature Engineering:
For NLP tasks, preprocess text data by tokenizing, stemming, or using techniques like TF-IDF or Word Embeddings.