**LAST MILE DELIVER BATCHING**

**Algorithm:**

1. **GA-SA Hybrid:**

Use Case: Dynamic Route Optimization with Rider Assignment

Characteristics:

- Combines the exploration ability of Genetic Algorithms (GA) with the escape-local-optima capability of Simulated Annealing (SA).

- Ideal for scenarios where routes need to be continuously optimized based on changing factors like new orders, traffic conditions, and rider availability.

Advantages:

- Efficiently explores diverse solutions through GA's crossover and mutation.

- SA's probabilistic acceptance helps escape local optima and adapt to changing conditions.

1. **Neural Networks (NN):**

Use Case: Order Identification for Rule-Based Assignment

Characteristics:

- Trained on historical data to identify patterns in orders meeting specific conditions (e.g., orders from the same kitchen to the same customer).

- Suitable for real-time order processing and categorization.

Advantages:

- Learns intricate patterns and relationships in data.

- Adaptable and capable of handling non-linear relationships.

Potential Implementation:

- Train a NN classification model to identify orders based on relevant features.

- Utilize the model for real-time prediction during order processing.

1. Random Forest:

Use Case: Rider Assignment Decision Making

Characteristics:

- Ensemble learning method that combines multiple decision trees.

- Well-suited for classification tasks and decision-making scenarios.

Advantages:

- Handles complex decision-making scenarios.

- Provides feature importance for transparency.

Flexibility:

- GA-SA Hybrid: Highly flexible, especially in scenarios with evolving conditions.

- NN: Flexible for pattern recognition but might need retraining for significant changes.

- Random Forest: Flexible and adaptable to various decision-making scenarios.

Real-time Adaptability:

- GA-SA Hybrid: Adaptable in real-time to dynamic changes.

- NN: Real-time predictions but may require periodic retraining.

- Random Forest: Real-time decision-making without frequent retraining.

- Training Complexity:

- GA-SA Hybrid: Requires careful tuning of parameters.

- NN: Requires tuning and regularization but is effective for non-linear relationships.

-Random Forest: Less sensitive to hyper parameter tuning and can handle non-linear decision boundaries effectively.

## In summary, the GA-SA hybrid is suitable for dynamic route optimization, NN for order identification, and Random Forest for decision-making scenarios in last-mile delivery. RULE #1: Delivery for Two Orders from Same Kitchen, Same Customer:

**Scenario:** Two orders (A, B) from Kitchen K1 to Customer C1, ready at 12:00 PM & 12:10 PM.

**Methods:**

1. **GA-SA Hybrid:**
   * Explores different routes, assigns riders.
   * Fine-tunes routes to minimize delivery time.
   * **Outcome:** Both orders assigned to the same rider for optimization.
2. **Neural Networks (NN):**
   * Trained on historical data to recognize patterns.
   * Predicts if orders meet rule conditions.
   * **Outcome:** Predicts both orders meet rule, suggesting same rider assignment.
3. **Random Forest:**
   * Trained on historical data to assign riders based on features.
   * Suggests best rider for each order.

## Outcome: Suggests same rider for both orders for optimal delivery.

## Rule #2: Two Orders, Different Kitchens, Same Customer (Opposite Ends)

**Scenario:** Two orders, different kitchens K1 & K2 (1km apart), same customer C1, ready at 12:00 PM & 12:10 PM. Assign to one rider.

**Methods:**

**1. GA-SA Hybrid:**

* Explores routes with rider handoff at a central point.
* Refines routes for minimal travel time and handoff efficiency.
* **Outcome:** One rider picks up, hands off at midpoint, another delivers to customer.

**2. Random Forest:**

* Trained to consider kitchen locations and handoff feasibility.
* Suggests best handoff point and rider assignment for minimal travel time.
* **Outcome:** Recommends efficient rider assignment and handoff strategy.

## Rule #3: Two Orders, Same Kitchen, Different Customers (1km apart):

**Scenario:** Two orders, same kitchen K1, different customers C1 & C2 (1km apart), ready at 12:00 PM & 12:10 PM.

**Methods:**

**1. GA-SA Hybrid:**

* Explores efficient routes for one rider to pick up both orders.
* Refines routes for minimal travel time.
* **Outcome:** Assigns one rider for optimal delivery to both customers.

**2. Neural Networks (NN):**

* Trained to recognize the pattern.
* Suggests optimal rider considering customer locations.
* **Outcome:** Recommends efficient rider assignment.

**3. Random Forest:**

* Trained to consider customer locations in rider assignment.
* Suggests best rider for minimal travel time.

## Rule #4: Two Orders, Different Kitchens, Same Customer (10 mins apart)

**Methods:**

**1. GA-SA Hybrid:**

* Explores routes to pick up both orders efficiently.
* Refines them for minimal travel time.
* **Outcome:** Assigns one rider for optimal delivery.

**2. Neural Networks (NN):**

* Trained to recognize the pattern.
* Suggests optimal rider considering kitchens' locations.
* **Outcome:** Recommends efficient rider assignment.

**3. Random Forest:**

* Trained to consider kitchen locations in rider assignment.
* Suggests best rider for minimal travel time.

## Rule #5: Two Orders, Different Kitchens, Same Customer (1km apart)

**Scenario:** Two orders, different kitchens K1 & K2 (1km apart), same customer C1, ready at 12:00 PM & 12:10 PM. Assign to one rider.

**Methods:**

**1. GA-SA Hybrid:**

* Explores efficient routes for one rider to pick up both orders.
* Refines routes for minimal travel time.
* **Outcome:** Assigns one rider for optimal delivery to the customer.

**2. Neural Networks (NN):**

* Trained to recognize the pattern.
* Suggests optimal rider considering kitchen locations.
* **Outcome:** Recommends efficient rider assignment.

**3. Random Forest:**

* Trained to consider kitchen locations in rider assignment.
* Suggests best rider for minimal travel time.

## Rule #6: Two Orders, Same Customer, 2nd Pick up on the Way (Ready at Arrival)

**Scenario:** Two orders for the same customer C1. Order B from kitchen K2 is in the route to C1 from kitchen K1 (Order A). Both orders are ready when the rider reaches K2.

**Methods:**

**1. GA-SA Hybrid:**

* Explores routes efficiently integrating K2 pickup into the K1-C1 route.
* Refines routes for minimal travel time.
* **Outcome:** Assigns one rider for optimal delivery, including K2 pickup.

**2. Neural Networks (NN):**

* Trained to recognize the pattern and route optimization opportunities.
* Suggests optimal rider and efficient route with K2 pickup integration.
* **Outcome:** Recommends efficient rider assignment and delivery strategy.

**3. Random Forest:**

* Trained to consider pickup order and location for optimization.
* Suggests best rider and route, prioritizing minimal travel time with K2 pickup.

## Rule #7: Multi-Stop Delivery (Drops & Pickups on the Way)

**Scenario:** Two orders :

* Second customer's drop on the way to the first (or vice versa).
* Second kitchen's pickup on the way to a customer.
* Both orders ready at the same time (10 mins apart or on arrival).

**Methods:**

**1. GA-SA Hybrid:**

* Explores routes efficiently integrating all stops (drops & pickups).
* Refines routes for minimal travel time.
* **Outcome:** Assigns one rider for optimal delivery, including kitchen pickup and order drops.

**2. Neural Networks (NN):**

* Trained to recognize complex delivery patterns and optimize routes.
* Suggests optimal rider and efficient route with all stops integrated.
* **Outcome:** Recommends efficient rider assignment and delivery strategy.

**3. Random Forest:**

* Trained to consider complex stop order and location for optimization.
* Suggests best rider and route, prioritizing minimal travel time with all stops.

## Rule #8: Same Kitchen, Multi-Stop Delivery (On-the-Way Drops)

**Scenario:** Two orders from the same kitchen, with:

* Second customer's drop on the way to the first (or vice versa).
* Both orders ready at the same time (10 mins apart).

**Methods:**

**1. GA-SA Hybrid:**

* Explores efficient routes integrating drops from the same kitchen.
* Refines routes for minimal travel time.
* **Outcome:** Assigns one rider for optimal delivery, including on-the-way drops.

**2. Neural Networks (NN):**

* Trained to recognize similar kitchen, multi-stop patterns.
* Suggests optimal rider and route with efficient drop integration.
* **Outcome:** Recommends efficient rider assignment and delivery strategy.

**3. Random Forest:**

* Trained to consider kitchen and drop locations for optimization.
* Suggests best rider and route, prioritizing minimal travel time with drops.