- Initial Instance (I): {A1 = 50,000 (Income), A2 = 650 (Credit Score), A3 = 4 (Years of Employment), A4 = 20,000 (Loan Amount), A5 = 0.25 (Debt-to-Income Ratio)}
- Scoring Function (fc):
  - p = [0,1]
  - If p >= 0.5 → C(I) = 1 , else C(I) =0;
- Decision-making System (C):
  - C(I) = 0 → Loan Rejected
  - C(I) = 1 → Loan Approved
- Initially,

  p = 0.3

  C(I)= 0 → Loan Rejected

  combinations = [{}, 0.3] (Empty set with score 0.3)

Iteration 1 (i = 0)

1. pop set with the smallest score from combinations:
   - combination = {} with score p = 0.3.
2. Counterfactual (no features changed):
   - counterfactual = I'({}).
3. Decision remains the same:
   - C(counterfactual) = C(I) = 0 (Loan Rejected).
4. Explore features:
   - Iterate over features, because explanation_list is empty:
     - For each feature Aj (e.g., A1 = 50,000, A2 = 650, etc.), check if it is not in the counterfactual set.
     - If not, add it to a new set E, compute the score fc(I'(E)), by replacing the features' value to their counterfactual values and insert E into combinations.
5. For example:
   - E = {A1} with score p = 0.55 (Insert {A1}, score 0.55 into combinations).
   - E = {A2} with score p = 0.50 (Insert {A2}, score 0.50 into combinations).
   - E = {A3} with score p = 0.65 (Insert {A2}, score 0.65 into combinations).
   - E = {A4} with score p = 0.45 (Insert {A2}, score 0.45 into combinations).
   - E = {A5} with score p = 0.51 (Insert {A2}, score 0.51 into combinations).
6. End of iteration 1:

- combinations = [{A1}, 0.55], [{A2}, 0.50], [{A3}, 0.65], [{A4}, 0.45], [{A5}, 0.51].

Iteration 2 (i = 1)

1. pop set with the smallest score from combinations:
   - combination = {A4} with score p = 0.45.
2. Counterfactual:
   - counterfactual = I'({A4}) (Set A4 to its counterfactual value, keeping the rest of the features as it is).
3. Decision remains the same:
   - C(counterfactual) = 0 (Loan Rejected).
4. Explore features:
   - Add features to the combination and compute scores, because explanation_list is empty:
     - E = {A4, A1} with score p = 0.58 (Insert {A4, A1}, score 0.58 into combinations).
     - E = {A4, A2} with score p = 0.50 (Insert {A4, A2}, score 0.50 into combinations).
     - E = {A4, A3} with score p = 0.65 (Insert {A4, A3}, score 0.65 into combinations).
     - E = {A4, A5} with score p = 0.49 (Insert {A4, A5}, score 0.49 into combinations).
     - Continue for all combinations.
5. End of iteration 2:
   - combinations = [{A1}, 0.55], [{A2}, 0.50], [{A3}, 0.65], [{A5}, 0.51], [{A4, A1}, 0.58], [{A4, A2}, 0.50], [{A4, A3}, 0.65], [{A4, A5}, 0.49].

Iteration 3 (i = 2)

1. pop set with the smallest score from combinations:
   - combination = {A4, A5} with score p = 0.49.
2. Counterfactual:
   - counterfactual = I'({A4, A5}) (Set A4 and A5 to its counterfactual value, keeping the rest of the features as it is).
3. Decision remains the same:
   - C(counterfactual) = 0 (Loan Rejected).
4. Explore features:
   - Add features to the combination and compute scores, because explanation_list is empty:

        - E = {A4, A5, A1} with score p = 0.74 (Insert {A4, A5, A1}, score 0.74 into combinations).
        - E = {A4, A5, A2} with score p = 0.64 (Insert {A4, A5, A2}, score 0.64 into combinations).
        - E = {A4, A5, A3} with score p = 0.54 (Insert {A4, A5, A3}, score 0.54 into combinations).
5. End of iteration 3:
    - combinations = [{A1}, 0.55], [{A2}, 0.50], [{A3}, 0.65], [{A5}, 0.51], [{A4, A1}, 0.58], [{A4, A2}, 0.50], [{A4, A3}, 0.65], [{A4, A5, A1}, 0.74], [{A4, A5, A2}, 0.64], [{A4, A5, A3}, 0.54].

Iteration 4 (i = 3)

1. pop set with the smallest score(two such sets are available, hence one is chosen randomly) from combinations:
    - combination = {A4, A2} with score p = 0.50.
2. Counterfactual:
    - counterfactual = I'({A4, A2}) (Set A4 and A2 to its counterfactual value, keeping the rest of the features as it is).
3. Decision changes:
    - C(counterfactual) = 1 (Loan Approved).
4. Combination is causal:
    - The combination {A4, A2} is causal
    - explanation=combination={A4, A2}
5. Irreducibility:
    - Now, check for each set 'E' in power set of combination → {{},{A4},{A2},{A4, A2}}:
        - C(I'{})= 0.3 -> No change in feature, results in loan rejection
        - C(I'{A4})= 0.45 -> changing A4 to its counterfactual value results in loan rejection
        - C(I'{A2})=0.50 -> changing A2 to its counterfactual value results in loan approval & E is smaller than explanation

          explanation = E = {A2}

        - C(I'{A4, A2})=0.50 -> changing A4 & A2 to its counterfactual value also results in loan approval, but here E is larger than explanation, so no change in explanation variable.
6. Final Explanation list:
    - The explanation is {A2} added to explanation_list.
    - explanation_list = {A2}

7. End of iteration 4:
    - combinations = [{A1}, 0.55], [{A2}, 0.50], [{A3}, 0.65], [{A5}, 0.51], [{A4, A1}, 0.58], [{A4, A3}, 0.65], [{A4, A5, A1}, 0.74], [{A4, A5, A2}, 0.64], [{A4, A5, A3}, 0.54].

Note: Here, in algorithm there should be a logic to remove explanation_list items and also its supersets from combinations to reduce computational cost for re-calculating the counterfactual values and scores further.

Iteration 5 (i = 4)

1. pop set with the smallest score from combinations:
    - combination = {A2} with score p = 0.50.
2. Counterfactual:
    - counterfactual = I'({A2}) (Set A2 to its counterfactual value, keeping the rest of the features as it is).
3. Decision changes:
    - C(counterfactual) = 1 (Loan Approved).
4. Combination is causal:
    - The combination {A2} is causal
    - explanation=combination={A2}
5. Irreducibility:
    - Now, check for each set 'E' in power set of combination → {{},{A2}}:
        - C(I'{})= 0.3 -> NO counterfactual value results in loan rejection
        - C(I'{A2})=0.50 -> changing A2 to its counterfactual value results in loan approval & E is equal to explanation
6. Final Explanation list:
    - The explanation is {A2} added to explanation_list.--> already there, so no changes
    - explanation_list = {A2}
7. End of iteration 5:
    - combinations = [{A1}, 0.55], [{A3}, 0.65], [{A5}, 0.51], [{A4, A1}, 0.58], [{A4, A3}, 0.65], [{A4, A5, A1}, 0.74], [{A4, A5, A2}, 0.64], [{A4, A5, A3}, 0.54]

Iteration 6 (i = 5)

1. pop set with the smallest score from combinations:
    - combination = {A5} with score p = 0.51.
2. Counterfactual:
    - counterfactual = I'({A5}) (Set A5 to its counterfactual value, keeping the rest of the features as it is).

3. Decision changes:
    - C(counterfactual) = 1 (Loan Approved).
4. Combination is causal:
    - The combination {A5} is causal
    - explanation=combination={A5}
5. Irreducibility:
    - Now, check for each set 'E' in power set of combination → {{},{A5}}:
        - C(I'{})= 0.3 -> NO counterfactual value results in loan rejection
        - C(I'{A5})=0.51 -> changing A5 to its counterfactual value results in loan approval & E is equal to explanation
6. Final Explanation list:
    - The explanation is {A5} added to explanation_list.
    - explanation_list = [{A2},{A5}]
7. End of iteration 6:

combinations = [{A1}, 0.55], [{A3}, 0.65], [{A4, A1}, 0.58], [{A4, A3}, 0.65], [{A4, A5, A1}, 0.74], [{A4, A5, A2}, 0.64], [{A4, A5, A3}, 0.54].

Iteration 7 (i = 6)

1. pop set with the smallest score from combinations:
    - combination = {A4, A5, A3} with score p = 0.54.
2. Counterfactual:
    - counterfactual = I'({A4, A5, A3}) (Set A4, A5, A3 to its counterfactual value, keeping the rest of the features as it is).
3. Decision changes:
    - C(counterfactual) = 1 (Loan Approved).
4. Combination is causal:
    - The combination {A4, A5, A3} is causal
    - explanation=combination={A4, A5, A3}
5. Irreducibility:
    - Now, check for each set 'E' in power set of combination → {{},{A4},{A5},{A3},{A4,A5},{A5,A3},{A4,A3},{A4, A5, A3}}:
        - C(I'{})= 0.3 -> NO counterfactual value results in loan rejection
        - C(I'{A4})=0.45 -> changing A5 to its counterfactual value results in loan rejection
        - C(I'{A5})=0.51 -> changing A5 to its counterfactual value results in loan approval & E is smaller than explanation

          explanation = E = {A5}

Note: This A5 is already explored in the previous iteration, but because it was not removed from all available combinations, it had to be explored again by the algorithm.

- C(I'{A3})=0.65 -> changing A5 to its counterfactual value results in loan approval & E is equal to size of explanation

Note: If we would have explored this E={A3} prior to E={A5}, we would have assigned this {A3} to the explanation variable and hence finally leading to the addition of A3 into explanation_list. Hence the ordering of power set elements plays a deciding factor. But if all the superset of explanation_list items would have been deleted from combinations, then this case would have been addressed efficiently.

- C(I'{A4, A5})=0.49 -> changing A5 to its counterfactual value results in loan rejection
- C(I'{A5, A3})=0.48 -> changing A5 to its counterfactual value results in loan rejection
- C(I'{A4, A3})=0.65 -> changing A5 to its counterfactual value results in loan approval & E is larger than explanation
- C(I'{A4,A5,A3})=0.54 -> changing A5 to its counterfactual value results in loan approval & E is larger than explanation

6. Final Explanation list:
   - The explanation is {A5} added to explanation_list→ already exists.
   - explanation_list = [{A2},{A5}]
7. End of iteration 7:

combinations = [{A1}, 0.55], [{A3}, 0.65], [{A4, A1}, 0.58], [{A4, A3}, 0.65], [{A4, A5, A1}, 0.74], [{A4, A5, A2}, 0.64].

Final explanation_list till iteration 7:

explanation_list = [{A2},{A5}]