ITU Artificial Intelligence & Data Engineering Department
YZV 202E Optimization for Data Science, Spring 2023
Term Project
Project Team Registration Due: May 12, 2023 11:59pm
Project Deliveries Due: May 29, 2023 11:59pm

1. General Project Description

You will apply appropriate design methods to develop a solution or prototype based on a selected optimization technique. You need to implement a method in a Jupyter Notebook for the problem, where it needs to have some constraints or the method's hyperparameters should be interactively adjustable using sliders or text boxes.

2. Deliverables

1. **PDF Report:** Prepare a report in IEEE conference format, including sections describing the problem, formal method formulation, working mechanics explanation, real-world application, experimental evaluation, and conclusion.
2. **Jupyter Notebook:** Provide an executable Jupyter Notebook with your implementation, experiments, and adjustable hyperparameters using sliders or text boxes.
3. **Presentation:** Prepare a PDF or PPTX presentation that outlines the problem, method, experiments, and results.

5. Group Work and Collaboration

The project can be completed individually or in groups of two or three, depending on the complexity of the problem, modeling, and experimentation.

6. Policy on the Use of Existing Code

You are allowed to use GitHub code from research papers, but there must be a new application, problem, or dataset that you apply the method to.

7. Presentation and Grading

The projects will be presented in class (online via Zoom) during the final week of the semester. The grading will be as follows:

1. **Presentation (%20):** Clear and concise explanation of the problem, method, experiments, and results.
2. **Report (%40):** Well-structured problem description, method presentation, experimental evaluation, and conclusion.
3. **Code, Simulation, and Experimentation (%40):** Functioning code, adjustable hyperparameters using sliders or textboxes, and effective experimentation.

8. Project Alternatives

Below we have project description and guidelines for three alternative projects:

1. Post-earthquake relief supply distribution in Turkey.
2. However, you can tackle a real-world optimization problem related to public health, safety, welfare, or environmental, cultural, social, and economic factors. You may choose to work on the real problem below or work on a similar problem concerning Turkey.
3. Exploring alternative optimization methods for training neural networks.

By the deadline you must register your project title and team.

8.1. Post-earthquake relief supply distribution in Turkey

In this project, you will tackle a real-world optimization problem related to public health, safety, welfare, or environmental, cultural, social, and economic factors. You may choose to work on the real problem below or work on a similar problem concerning Turkey.

8.1.1. Description

A powerful earthquake has hit Turkey, causing significant damage to 11 cities and resulting in more than 50,000 deaths. Some of the cities' road connections have been blocked for a few days, and there are immediate petrol and supply shortages. The government and relief organizations need to efficiently distribute food, water, medicine, and other essential supplies to the affected cities while considering transportation challenges and fuel shortages.

8.1.2. Characteristics

Turkey has 11 affected cities, each with a specific demand for relief supplies based on their population and the extent of the damage. There are multiple distribution centers across Turkey, located in unaffected cities or areas, with varying supplies. The transportation costs and time between distribution centers and affected cities depend on the distance, road conditions, and availability of petrol. Supplies may be perishable, with some items having a limited shelf life.

Some cities have road access blocked for a few days, requiring alternative transportation methods such as helicopters or boats, which may have different transportation costs and capacities. Randomness can be introduced to the problem by having variable demands and supplies, reflecting the uncertainty in the actual needs and availability of resources in a post-disaster scenario. Objective: Determine an optimal distribution plan that allocates supplies from distribution centers to the affected cities, minimizing the total transportation costs and response time while satisfying the demands of each city and considering the transportation challenges and petrol shortages.

8.1.3. Example possible solution approaches

(i) Genetic Algorithms: Encode the distribution plan as a chromosome and use genetic operators such as mutation and crossover to explore the solution space. The fitness function would be the total transportation costs and response time, considering the constraints related to blocked roads and fuel shortages.

(ii) Particle Swarm Optimization: Represent each potential solution as a particle in the swarm, and update the particles iteratively based on the social and cognitive components to minimize the objective function, taking into account the constraints.

(iii) Simulated Annealing: Design a random search algorithm that accepts or rejects solutions based on a temperature parameter. Decrease the temperature over time to converge to an optimal solution while considering the unique constraints of the problem.

(iv) Mixed-Integer Linear Programming: Formulate the problem as a mixed-integer linear programming (MILP) problem, with appropriate constraints to represent the supplies, demands, transportation costs, and fuel shortages. Apply MILP solvers to find the optimal distribution plan.

(v) Extensions of the mentioned methods: Combine or modify the mentioned methods to create hybrid algorithms or develop problem-specific heuristics for more efficient solutions, taking into account the unique constraints of the Turkey earthquake scenario.

By using these optimization methods, you can develop a solution to efficiently distribute the supplies in the post-earthquake recovery scenario in Turkey, considering the transportation challenges, fuel shortages, and randomness in demand and supply.

8.1.4. Your Approach

(i) You will create your own problem description and simulation.

(ii) Compare two different algorithms to solve the same problem.

(iii) Visualizations on maps are highly encouraged.

(iv) Interactive demand posting with interactive forms or any other idea which makes the tool more useful can be incorporated (e.g., polling from social media).

(v) Use real or realistic data for costs, distances, times, traffic, earthquake or city locations is required for full experimentation design points. However, you may still work with completely random data.

(vi) You can use existing datasets over the net from IBB, Kaggle, WHO, NASA etc.

You can select appropriate optimization methods that satisfy the constraints and requirements.

8.2. Reshaping the problem to other domains

You can reshape the problem to work on floods, fires, crime, water supply, schools etc.

8.3. Exploring Alternative Optimization Methods for Training Neural Networks (Limited to first registered 10 projects only)

In this project, you will explore alternative optimization methods for training neural networks, focusing on gradient-free techniques such as Particle Swarm Optimization (PSO), Genetic Algorithms (GA), Bayesian Optimization (BO), and Simulated Annealing (SA). You will research how these methods can be used individually or in combination with gradient methods like Gradient Descent to improve neural network training. Your project must present a comparison of methods on a well-known benchmark dataset.

This task involves the steps mentioned below.

8.3.1. Literature Review and Method Selection

Check the example research papers and additional sources to gain a deeper understanding of alternative optimization methods for neural network training. Select one or more gradient-free optimization methods to investigate, such as PSO, GA, or SA, BO.

8.3.2. Method Implementation and Experimentation

Implement the chosen optimization method(s) for training neural networks. Apply the method(s) to a well-known benchmark dataset (if the paper already supplies a wellknown benchmark test and code, you must test it on a new benchmark test) and compare the performance with traditional gradient-based techniques like Gradient Descent.

8.3.3. Method Comparison and Analysis

Analyze the performance of each chosen gradient-free method and compare their strengths and weaknesses. Investigate the potential benefits of combining gradient-free methods with gradient methods for neural network training.

8.3.4. Deliveries

Your PDF Report and presentation should also involve the sections describing the literature review, chosen optimization method(s).

8.3.5. References

Check the example papers below. You can select one, suggest another or suggest your own algorithm.

1. H. Kim et al., ”Variable three-term conjugate gradient method for training artificial neural networks”, *Neural Networks* 159, 2023. `https://doi.org/10.1016/j.neunet.2022.12.001` [CG, NN]

2. H. Iiduka and Y. Kobayashi, ”Training Deep Neural Networks Using Conjugate Gradient-like Methods”. *Electronics* 9(11) p. 1809, 2020. `https://doi.org/10.3390/electronics9111809` [CG, NN]

3. I. G. Tsoulos, A. Tzallas, E. Karvounis, and D. Tsalikakis, ”Bound the Parameters of Neural Networks Using Particle Swarm Optimization,” *Computers* 12(4) p. 82, 2023. `https://doi.org/10.3390/computers12040082` [PSO, NN]

4. J.-R. Zhang, J. Zhang, T.-M. Lok, and M. R. Lyu, ”A hybrid particle swarm optimization–back-propagation algorithm for feedforward neural network training,” *Applied Mathematics and Computation* 185(2) pp. 1026-1037, 2007. `https://doi.org/10.1016/j.amc.2006.07.025` [PSO, BP]

5. F. E. F. Junior and G. G. Yen, ”Particle swarm optimization of deep neural networks architectures for image classification,” *Swarm and Evolutionary Computation* 49 pp. 62-74, 2019. `https://doi.org/10.1016/j.swevo.2019.05.010` [PSO, BP]

6. C. L. Kuo, E. E. Kuruoglu, and W. K. V. Chan, ”Neural Network Structure Optimization by Simulated Annealing,” *Entropy* 24(3), p. 348, 2022. `https://doi.org/10.3390/e24030348` [SA, NN]

7. S. Li, W. Xing, R. M. Kirby, and S. Zhe, ”Multi-Fidelity Bayesian Optimization via Deep Neural Networks,” In Advances in neural information processing systems, pp. 8521-8531, 2020. `https://proceedings.neurips.cc/paper/2020/hash/60e1deb043af37db5ea4ce9ae8d2c9ea-Abstract.html` [BO, NN]

8. D. Eriksson, M. Pearce, J. Gardner, R. D. Turner, and M. Poloczek, ”Scalable Global Optimization via Local Bayesian Optimization,” In Advances in neural information processing systems, 2019. `https://proceedings.neurips.cc/paper_files/paper/2019/hash/6c990b7aca7bc7058f5e98ea909e924b-Abstract.html` [BO, NN]