EAS Assignment Brief

CS3910 Computational Intelligence	Computational Intelligence Solution for Logistics
Anikó Ekárt a.ekart@aston.ac.uk Harry Goldingay h.j.goldingay1@aston.ac.uk	https://wass.aston.ac.uk/pages/viewcalendar.page.php?makeapp=1&cal_id=109 https://wass.aston.ac.uk/pages/viewcalendar.page.php?cal_id=469

Coursework Context

There are many large logistics transport networks (for example Palletways) that provide cost-effective express collection and delivery services to private and business customers. In such a network tens of thousands of orders are collected from their source and delivered to their destination on a daily basis. To reduce their environmental impact and their cost of operation at the same time it is important for transport networks to optimise their vehicle usage.

Orders usually consist of so-called pallets of predefined size and allowed maximum weight and the palletised goods are usually loaded on a truck as shown in Fig 1.a. Such networks often operate a hub-and-spoke model (Fig 1.b.), where the haulage companies (also called spokes) take their own customers' goods to a centralised hub where they are unloaded for delivery by other spokes and then load their lorries with goods from other spokes that are taken back to their own area for delivery. The journey of an order from source to destination (as shown in Fig 1.c.) starts with the order being picked up from the customer (source), then the goods are taken to the local depot, from where they are transported to a central hub. At the hub the trucks are unloaded, pallets are sorted and reloaded onto trucks corresponding to the haulage companies who deliver in the area of their destination. Then the pallets are delivered to the destination area's local depots and from there to their final destination.

The haulage companies' delivery areas are joined up to ensure their combination completely covers the required distribution area of the network. Efficiency gains are obtained by enabling haulage companies to accommodate customer deliveries to anywhere in the network while only having to cover journeys within their own delivery area and to and from the hubs. Resources are still under-utilised, as there can be cases when a truck runs completely empty or has substantial empty space.

It is this area of "transporting air" while still ensuring that everything is delivered on time that forms the context of this coursework.

Fig.1.a Pallets on a truck b. Hub and spoke model c. The journey of a pallet from source to destination

You can read more about the context here:

[1] E. Ilie-Zudor, A. Ekárt, Zs. Kemeny, C. Buckingham, P. Welch and L. Monostori , (2015),"Advanced predictive-analysis-based decision support for collaborative logistics networks", Supply Chain Management: An International Journal, Vol. 20 Issue 4 pp. 369 – 388. http://dx.doi.org/10.1108/SCM-10-2014-0323

More specifically, you have a chance to contribute to the more optimal allocation of trucks for one haulage company. The problem is defined as the optimisation of the number of trucks to be sent to the hub, based on the expected number of pallets for the return journey (i.e. pallets to be picked up from hub and delivered in own local area). It is known that each truck can hold up to 35 pallets, therefore the number of trucks can easily be calculated if the demand is known. The demand is only known for sure at the end of a day (at the hub), while the number of trucks has to be decided based on the estimation that can be made at 12pm.

Assignment Brief

You have the chance to architect a computational intelligence solution that minimises the error when estimating this demand. **The aim is to find a way of combining a set of provided values, such that the overall error is minimised over a period of time.**

You need to choose an appropriate computational intelligence approach and then go through the stages discussed in the lectures and practiced in the labs:

- Create an appropriate representation for candidate solutions (explained below under developing candidate solutions).
- Apply the evaluation function (explained below).
- Design and implement the operators.
- Design and implement a run of your algorithm.
- Experiment with various parameter settings, operators to improve the quality of your results.
- Investigate, evaluate and report on the ability of your approach to solve the problem, including comparison of different parameter settings and/or variants of your algorithm.

You are provided with historical data over a period of two months (working days only), in the file `cwk_train.csv`, where each row corresponds to one day. Each row in the file represents the following data:

- the known demand at the end of the day followed by 13 different measurements for the same day taken at 12pm.

You can test your solution on the historical data over the period of a consecutive month, provided in the file `cwk_test.csv`, using the same format. So, your input data is in columns 2-14, while the first column is the output data.

Your fitness evaluation needs to calculate the error, that is how far this value is from the actual known demand for each of the rows in the `cwk_train.csv` file and sum the errors, so if there are N rows in the file,

$$cost = \frac{1}{N}\sum_{i=1}^{N}|estimate_i - demand_i|,$$

where $estimate_i$ is the calculation of the estimate based on your candidate solution for row *i* and $demand_i$ is the known demand for row *i* provided in column 1 and | | is used to indicate absolute value.

You can then check how well your best solution is performing on the test data provided in the `cwk_test.csv` file.

You may choose any of the methods discussed in the lectures (i.e any evolutionary algorithm, particle swarm optimisation, ant colony optimisation, genetic programming).

> *Note that at the time of release, i.e. 18 October 2019, we have not covered evolutionary algorithms or genetic programming yet.*

You can apply the operators, methods, parameters exactly as provided in the lectures or you can choose to investigate others in the literature and create your own. **Note that the marking scheme includes an aspect dedicated to quality of solution.**

Developing candidate solutions

As stated above, your objective is to find some way to combine the 13 input measures to predict demand. You could use genetic programming to find the function that represents the best combination of the measurements to produce the estimate or you could design a parameterised model relating inputs and demand, and then optimise its parameters.

As an example of the latter approach you could, to take a simple example, assume that demand is just the sum of all input measures. Let $\vec{x_i} = (x_{i,1}, x_{i,2}, \dots, x_{i,13})$ be the input measures for some example i (i.e. a single row in one of the CSV files). Under this assumption, you would estimate demand as follows:

$$estimate_i = \sum_{j=1}^{13} x_{i,j}$$

There are two main problems with this model:

1. it has no flexibility – we have no way to tune its performance (and, therefore, nothing we can optimise using a CI algorithm),
2. it is not a good match for the observed demand.

You could address the first problem, and potentially the second, by introducing parameters – variables within the model which can be changed to allow estimates to better match the data. An example of how you could turn the original model into a parameterised model would be to change it from a sum into a weighted sum:

$$estimate_i = \sum_{j=1}^{13} w_j \cdot x_{i,j}$$

where $\vec{w} = (w_1, w_2, \dots, w_{13})$ are a set of numerical "weights" determining how much each of the input measures contributes to the estimate. You could then run a continuous/real-valued optimization algorithm to find the set of weights which minimizes the $cost$ function described above.

This is just an example model. You could use it for the coursework problem or try to develop your own based on your observations of relationships between variables in the test data. Remember that, if you do the latter, you will need to create a parameterised model or you will have nothing to optimise.

Descriptive details of Assignment:

The assignment consists of two parts: a lab sign-off to give you early feedback on your implementation and a final submission consisting of a report on your solution to the problem and associated source code.

- **Lab sign-off**: Based on completion of lab worksheets for Weeks 1-5. To be booked by the student, using WASS, expected to last 10-15 minutes. You should be prepared to demonstrate during the sign-off:

    1. The TSP problem, including a correct cost/fitness function,
    2. Random search implemented on the TSP,
    3. Local search implemented on the TSP,
    4. One computational intelligence algorithm of your choice implemented on the TSP,
    5. One computational intelligence algorithm of your choice implemented on the antenna array design problem.

    At the start of the sign-off appointment, we will provide you with **new test problem instances for both problems**, and your software should run on these. For the TSP, this will take the form of a CSV file of the same form as the ulysses16.csv file that you have already been working with. For the antenna array design problem, we will provide the number of antennae and steering angle for you to plug into your fitness function.

- **Software source code** file title (in a programming language of your choice): *Firstname_Lastname_CS3910_CI_solution.zip*

- **Report** title: *Firstname_Lastname_CS3910_CI_solution*, with preferred submission format as a single doc, rtf, docx or pdf file

- The report is limited to at most 1000 words and can include additional tables or graphs presenting results, up to 6 pages in total. The report should be typeset and presented in A4 format, using any easy to read font (for example Arial), size 12.

- Preferred reference style: Harvard referencing

Recommended reading/ online sources:

- References to additional academic articles on applying computational intelligence methods to similar problems can be provided upon request, but *they are not necessary for the completion of this coursework.*

- On Blackboard, you will find all relevant lecture material in the **Lecture material** folder within **Learning Resources** and all relevant lab material in the **Lab material** folder within **Learning Resources**.

Key Dates:

18/10/2019	Coursework set
23/10/2019	Supporting lecture regarding assessment
28/10/2019-1/11/2019 or 11/11/2019-15/11/2019 or 18/11/2019-23/11/2019	Sign-off session, to be booked by each student as a single individual office hour appointment of 15 minutes with either of the tutors
12/12/2019	Submission date for report and source code
16/01/2020	Expected feedback return date. Individual Feedback will be provided on Blackboard.
22/02/2020	Panopto recording on common good aspects of solutions and common mistakes made available

Submission Details:

- Sign off: book session as described above
- Source code: Online on Blackboard
- Report: Online on Blackboard

[Optional: Checklist to help students complete multifaceted submissions]

Marking Rubric:

The marking will be based on five aspects, described below.

#	**Aspect**	**Weight**	**Marks for aspect**	**Descriptor**
1	Sign-off	10	0 for a task	For the given task, the implementation does not behave as expected.
			10 for a task	For the given task, the implementation shows behaviour close to that expected, with minor flaws only.
			20 for a task	For the given task, the implementation behaves as expected.
2	Explanation and justification of solutions (report)	10	0-29	There is little or no evidence of justification for the solutions
			30-39	There is some explanation and justification, but below the threshold requirement for the module.
			40-49	There is a relevant explanation and justification, but it is limited in scope and depth.
			50-59	The explanation and justification are of sufficient depth and detail. Appropriate references are used where relevant.
			60-69	Substantial and well-presented explanation and justification of solutions.
			70-79	Thorough and in-depth explanation and justification of solutions.
			80-100	Excellent explanation and justification. Claims are fully supported with well chosen references.
3	Solution implementation (code)	30	0-29	There is little evidence of a computational intelligence implemented.
			30-39	There is some code that is relevant, however it contains bugs and/or is below the threshold requirement for the module.
			40-49	The code compiles and runs without errors. There is at least one solution implemented of some relevance to computational intelligence, but limited in scope and quality.
			50-59	There is at least one solution of sufficient scope and quality.
			60-69	There are two solutions of sufficient scope and quality.
			70-79	Very good solution implementation for two solutions, showing application of programming knowledge and software engineering principles.
			80-100	Excellent solutions implemented.

4	Quality of proposed solutions (report)	30	0-29	The solutions are of very limited use for the given problem.
			30-39	The solutions are of some use, however their quality is below the threshold requirement for the module.
			40-49	The solutions are of satisfactory quality, but are overly simple and lacking in scope.
			50-59	The solutions are of sufficient quality and complexity.
			60-69	Solutions indicating thinking beyond book-standard solutions.
			70-79	Very good solutions demonstrating some novelty or insight.
			80-100	Excellent solutions, demonstrating high levels of novelty and insight.
5	Evaluation of solutions (report)	20	0-29	There is little or no evidence of evaluation.
			30-39	There is some evidence of evaluation, but below the threshold for the module.
			40-49	There is some evidence of meaningful evaluation of quality of solutions.
			50-59	There is evidence of evaluation of quality of solutions, including comparison of the two solutions.
			60-69	The evaluation is systematic and is following the methods taught in the module.
			70-79	The evaluation and comparison have been carefully designed and allow for meaningful conclusion about the comparative quality of the solutions.
			80-100	In addition to the above, the insights gained from the evaluation are substantial.