The Machine Learning and Data Analysis (MLDA) Problem Set, v1.0

accompanying the GECCO 2019 Workshop on
Understanding Machine Learning Optimization Problems (UMLOP)

Workshop Organizers

- Pascal Kerschke, University of Münster, Münster, Germany
  kerschke@uni-muenster.de,
  `http://erc.is/p/kerschke`
- Marcus Gallagher, University of Queensland, Brisbane, Australia
  marcusg@uq.edu.au,
  `http://staff.itee.uq.edu.au/marcusg/`
- Mike Preuss, Leiden University, Leiden, The Netherlands
  m.preuss@liacs.leidenuniv.nl
  `https://www.universiteitleiden.nl/en/staffmembers/mike-preuss`
- Olivier Teytaud, Facebook AI Research, Paris, France
  olivier.teytaud@gmail.com

Abstract

This report presents the definitions of the problem set used for the GECCO 2019 workshop on *Understanding Machine Learning Optimization Problems (UMLOP)*, which is the successor of the PPSN 2018 workshop on *Investigating Optimization Problems from Machine Learning and Data Analysis*.

Introduction

Black-box optimization is a problem of major focus in the areas of evolutionary computation, metaheuristics and nature-inspired algorithms. Despite a large amount of research, there are relatively few standard benchmark problem sets available. Benchmark sets that are both based on real-world problems and well-suited to benchmarking are particularly rare.
In this report we propose the development of a set of benchmark optimization problems from the area of machine learning and data analysis. This could be considered a preliminary version of a benchmark set in this area, which we hope to develop further in the near future.

You can download the data needed when dealing with our test problems, by clicking on the links below:

- data for test problem 1
- data for test problem 2
- data for test problem 3
- data for test problem 4

For further information regarding this workshop, please refer to our workshop's website

`http://www.erc.is/go/gecco2019`

or send us an e-mail.

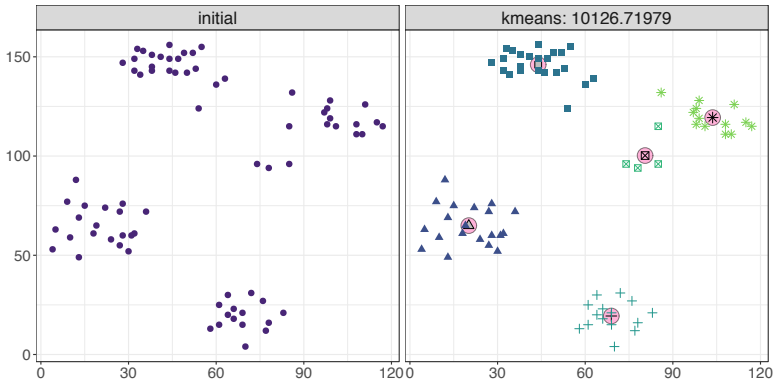
Test Problem 1: Sum of Squares Clustering Problems

The (continuous) sum of squares clustering problem provides the theoretical counterpart of the *facility location problem*. Given a set $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\} \subseteq \mathbb{R}^d$ of n data points (in the d-dimensional search space), determine the locations of a set of k cluster centers $\mathcal{C} = \{\mathbf{c}_1, \ldots, \mathbf{c}_k\} \subseteq \mathbb{R}^d$ such that

$$f(\mathcal{C}|\mathcal{X}) = \sum_{i=1}^{n} \sum_{j=1}^{k} b_{i,j} \cdot ||\mathbf{x}_i - \mathbf{c}_j||^2 \qquad \text{with} \qquad b_{i,j} = \begin{cases} 1, & \text{if } ||\mathbf{x}_i - \mathbf{c}_j||^2 = \min\limits_{l \in \{1,\ldots,k\}} ||\mathbf{x}_i - \mathbf{c}_l||^2 \\ 0, & \text{otherwise} \end{cases}.$$

is minimized. Here $||.||$ is the Euclidean distance metric (also known as L_2-norm) and the variables are the coordinates of the cluster centers in the data space. Denote the d-dimensional coordinates of the i-th cluster center as $\mathbf{c}_i = (y_{i+1}, \ldots, y_{i+d})$, then the problem can be rewritten as an unconstrained, continuous optimization problem of dimensionality $p = d \cdot k$, whose goal is finding the optimal p-dimensional vector $\mathbf{y}_{opt} = (\mathbf{c}_1, \ldots, \mathbf{c}_k)^T \in \mathbb{R}^p$.
A specific clustering problem instance is therefore defined by a dataset $\mathcal{X}$, and the desired number of clusters k. The dimensionality of the underlying optimization problem grows in proportion to d and k, but is independent of the number of data points n.

Problem 1a: Sum of Squares Clustering on the Ruspini data set. This is a two-dimensional dataset ($d = 2$) and we are interested in the optimal location of $k = 5$ cluster centers. Therefore, we look for $\mathbf{y}_{opt} \in \mathbb{R}^{10}$. The figure below illustrates the location of all $n = 75$ observations from the dataset, as well as the best solution found by means of running k-means.



The global optimum to this problem (10 126.71979) was reported in Du Merle et al. (2000) and corresponds to the following values as found via k-means:

$$\mathbf{y}_{opt} = \begin{pmatrix} \mathbf{c}_1 \\ \mathbf{c}_2 \\ \mathbf{c}_3 \\ \mathbf{c}_4 \\ \mathbf{c}_5 \end{pmatrix} = \begin{pmatrix} 20.150000000000 \\ 64.950000000000 \\ 43.913043478261 \\ 146.043478260870 \\ 68.933333333333 \\ 19.400000000000 \\ 80.500000000000 \\ 100.250000000000 \\ 103.615384615385 \\ 119.384615384615 \end{pmatrix}$$

Problem 1b: Sum of Squares Clustering on the German towns data set. This is a three-dimensional data set ($d = 3$) and the goal is to find the optimal locations for $k = 10$ cluster centers within the search space – or similarly, finding the global optimum of a $p = 30$ dimensional problem.

Possible variations for these problems could be the usage of alternative values for k, or randomly sampling datasets and thereby mimicking a "similar" problem instance. Note that these problems are in general unconstrained. However, reasonable solutions will have their cluster centers in the range of the dataset, so this can be used as bounds (e.g., for initializing the algorithms). The corresponding datasets, supporting `Matlab` code, the locations and the corresponding fitness values for the optimal locations of the cluster centers, as well as further information for these two problems can be found at: `http://realopt.uqcloud.net/ess_clustering.html`.

Test Problem 2: Multi-Layer Perceptron

Multi-layer perceptrons (MLP) are supervised learning models, which basically learn rules to map data from an input to an output space (see, e.g., Bishop, 1995). In recent years, their much larger, and hence more complex successors – deep learning neural networks – have drawn lots of interest and meanwhile form the state of the art in a variety of application areas, such as image analysis, speech recognition, cancer detection or self-driving cars. Yet, basically any classification or regression tasks with a reasonable amount of training and test data can be modeled with these networks. In order to get a better feeling for the structure and hence the decision process of such a network, the problems analyzed herein are based a "simple" 1-3-1 (MLP1) network (cf. Figure 1). These are fully connected feed-forward 1-D regression networks with one input node, three nodes in a single hidden layer, and an output node, complemented by biases and hyperbolic tangent (tanh) activation functions (in the hidden units). Given such a network, the problem then is to minimize

$$f(\boldsymbol{\theta}) = \frac{1}{n}\sum_{i=1}^{n}\left(g(x_i,\boldsymbol{\theta}) - y_i\right)^2 \qquad \text{with} \qquad g(x_i,\boldsymbol{\theta}) = v_0 + \sum_{j=1}^{3}\left(v_j \cdot \tanh(w_{0j} + x_i \cdot w_{1j})\right).$$

This is a 10-dimensional unconstrained continuous optimization problem over the set of parameters $\boldsymbol{\theta} = \{w_{01}, w_{11}, w_{02}, w_{12}, w_{03}, w_{13}, v_0, v_1, v_2, v_3\}$. The input data $x_1, \ldots, x_n \in [-1, 1]$ is made of $n = 50$ one-dimensional points, which are regularly spaced between -1 and 1. For the corresponding output y_i, four different problem sets, whose mappings are also depicted in Figure 2, are considered:

- P2a: quadratic function, i.e., $y_i = x_i^2$
- P2b: sine function, i.e., $y_i = \sin(x_i)$
- P2c: absolute value function, i.e., $y_i = |x_i|$
- P2d: heavyside function, i.e., $y_i = \begin{cases} 0, & \text{if } x < 0 \\ 1, & \text{if } x \geq 0 \end{cases}$.

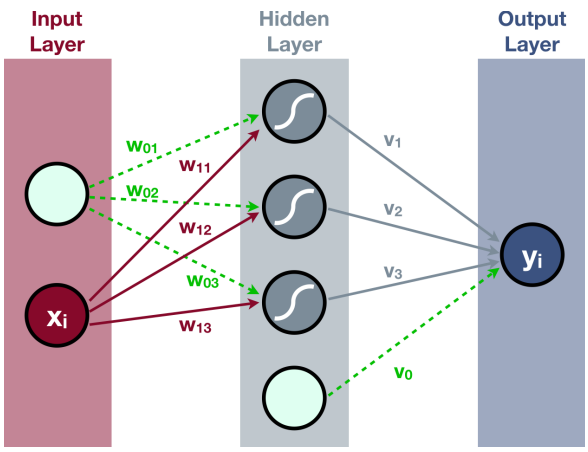


Figure 1: Schematic representation of the 1-3-1 (MLP1) network with one input node, three nodes in the (single) hidden layer and an output node.

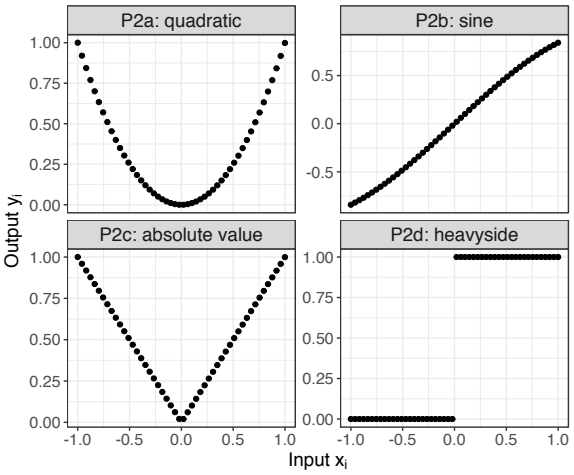


Figure 2: Mapping from input x_i to output y_i for each of the four problems: the quadratic (P2a), sine (P2b), absolute value (P2c) and heavyside (P2d) function (top left to bottom right).

Test Problem 3: Multidimensional Scaling via Sammon Mapping

Multidimensional scaling (MDS) methods attempt to find a low-dimensional representation of a given data set – e.g., to produce a 2- or 3-D visualization of the spatial distribution of the data points (see, e.g., Cox and Cox, 2000). One well-known criterion for this is known as Sammon mapping (Sammon, Jr., 1969), where the optimization problem is to minimize the Sammon stress function

$$E(\mathbf{z}_1, \ldots, \mathbf{z}_n) = \frac{1}{\sum_{r=1}^{n-1} \sum_{s=r+1}^{n} ||\mathbf{x}_r - \mathbf{x}_s||} \sum_{r=1}^{n-1} \sum_{s=r+1}^{n} \frac{\left(||\mathbf{z}_r - \mathbf{z}_s|| - ||\mathbf{x}_r - \mathbf{x}_s||\right)^2}{||\mathbf{x}_r - \mathbf{x}_s||}.$$

Here, $\mathbf{x}_r$ and $\mathbf{x}_s$ are a pair of points in the original (d-dimensional) data space, and $\mathbf{z}_r$ and $\mathbf{z}_s$ correspond to their low-dimensional representation. Further, $||.||$ is the Euclidean distance metric (also known as L_2-norm). The problem instances are defined by their data sets. So far, the high-dimensional data (i.e., $\mathbf{x}_1, \ldots, \mathbf{x}_n$) for two data sets are given (see below) and the goal is to find suitable sets of low-dimensional decision variables $\mathbf{z}_1, \ldots, \mathbf{z}_n$ that are optimal w.r.t. the aforementioned stress function.

Problem P3a: Ripley's Virus Dataset. Although the origins of this dataset date back to Fauquet et al. (1988), the data is more often referenced to Ripley (1996)[1], who used it to describe different multidimensional scaling techniques – including Sammon mapping – within his book. Here, we focus on the subset of the most frequent class of viruses within the original data: the so-called tobamoviruses. The corresponding dataset (`virus3.dat`) contains 38 observations of 18 measurements each and it is available at `https://www.stats.ox.ac.uk/pub/PRNN/`. Based on the amount of observations, finding a two-dimensional representation of this dataset transfers into a 76-dimensional unconstrained continuous optimization problem. Figure 3 shows the initial and final solution of an exemplary optimization run, as well as the trajectory of the Sammon stress function during the optimization process.

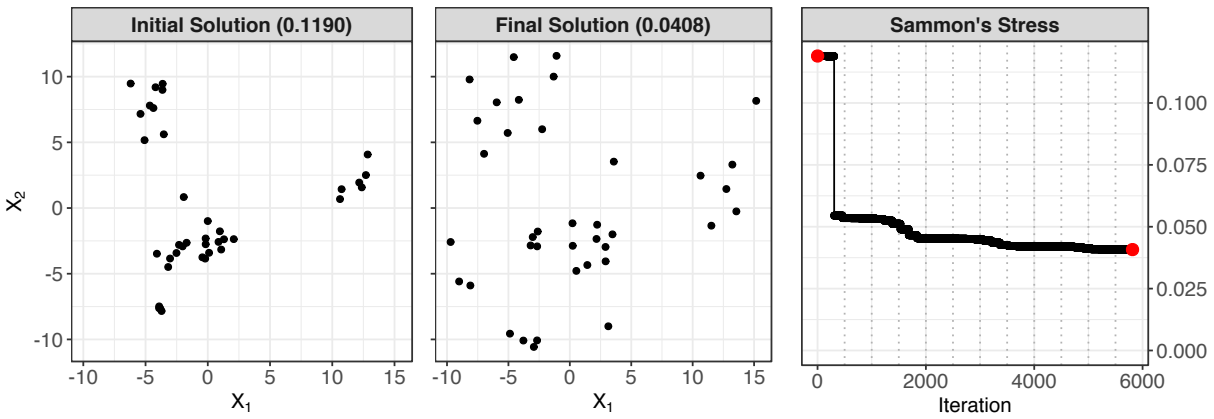


Figure 3: Visualization of the initial (left) and final solution (middle) of an exemplary optimization run on Problem P3a (Ripley's virus data). The image on the right displays the trajectory of the Sammon stress function during the optimization process.

Problem P3b: Lloyd's Bank Employee Data. This dataset was first published by Izenman (2008). It contains sequential employment records for 80 randomly selected employees from 1905 to 1909 and thus corresponds to a 160-dimensional unconstrained continuous optimization problem. The dataset is available online at: `https://astro.temple.edu/~alan/MMST/datasets.html`. The file (`samp05.xls`) contains an ID variable (ID), a variable recording the first year of the employee's employment (YEAR), and 71 variables containing the sequential data (V1-V71). Further details on this dataset, as well as a file with a proximity matrix (`samp05d.xls`) containing pre-calculated values of $||\mathbf{x}_r - \mathbf{x}_s||$, can also be found on the aforementioned website.

[1] Some further information can also be found here: `http://www.stats.ox.ac.uk/~ripley/MultAnal_HT2008/Viruses.pdf`

Test Problem 4: Landscape of the Planet Earth

A *Global Digital Elevation Map (GDEM)* image of the Earth (see Figure 4) is available online at `https://asterweb.jpl.nasa.gov/gdem.asp`. Consider this as a two-dimensional continuous optimization problem, where the goal is to find the maximum elevation value, $f(\mathbf{x})$ with $\mathbf{x} \in \mathbb{R}^2$. The resolution of the image is 4320×2160 pixels, which implies that this is a box-constrained optimization problem with $(x_1, x_2) \in [0, 4320] \times [0, 2160]$ being the feasible region for the search space.
Note that f defines a high-resolution staircase-like objective function, where the gradient is everywhere either zero or infinite. The elevation function f is implemented as a look-up table and the underlying data (4MB .zip file, 146MB uncompressed) is available online at: `https://drive.google.com/open?id=1EC7DzaQaw7jp3BDYnOIqQpdA4HpfnBcL`.

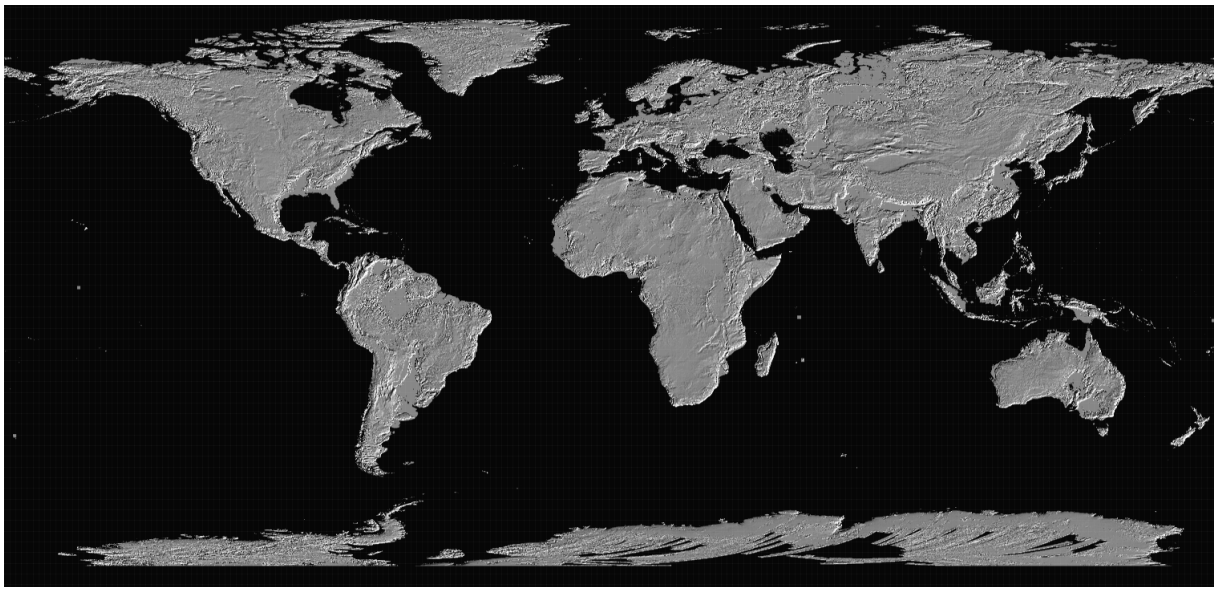


Figure 4: Visualization of the Earth's landscape.

Test Problem 5: Buoy Placement

This problem relies on a simulator, which computes the energy landscapes resulting from waves in the ocean. Those landscapes are the foundation for reasonable buoy placement, which in turn are used for (efficient) energy generation. Of course, it is desirable to place the buoys such that they produce as much energy as possible (cf. Figure 5). The *Optimisation and Logistics* group at the University of Adelaide provides a `Matlab`-based implementation[2] of such an energy landscape simulator.
While finding the ideal location of a single buoy might still be quite easy, positioning multiple buoys is a rather challenging task as every single buoy influences the waves and thereby changes the energy landscape. In addition, even single evaluations with this `Matlab`-based simulator are already time consuming and thus make this an expensive optimization problem. There also exist first analyses based on this simulator (Arbonès et al., 2016, 2018; Neshat et al., 2018; Wu et al., 2016), which of course might provide a good introduction into this topic.

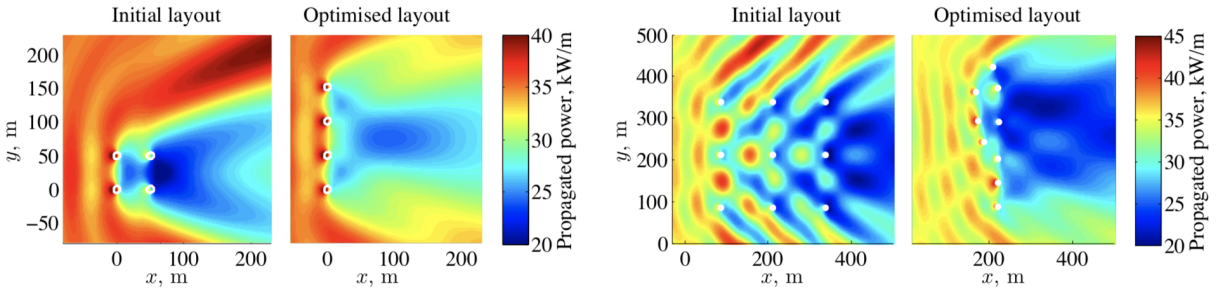


Figure 5: Exemplary placement of four (left) and nine (right) buoys, respectively. This figure was taken from Arbonès et al. (2018).

Here, we are interested in the ideal location(s) for $b \in \{1, 3, 5\}$ buoys in the ocean's two-dimensional surface (based on the simulator), which corresponds to 2-, 6- and 10-dimensional box-constrained optimization problems.
Aside from the aforementioned optimization problem, we also look for submissions which analyze the corresponding energy landscape(s), including how they change based on the buoy placement. Of course, faster implementations of the simulator (in any common programming language, such as `C`, `Java`, `python`, `R`, etc.) are highly appreciated as well.

[2] `https://cs.adelaide.edu.au/~optlog/research/wind/2016gecco-wec-code.zip`

References

Dídac Rodríguez Arbonès, Boyin Ding, Nataliia Y. Sergiienko, and Markus Wagner. Fast and Effective Multi-Objective Optimisation of Submerged Wave Energy Converters. In *Proceedings of the 14th International Conference on Parallel Problem Solving from Nature*, pages 675 – 685. Springer, September 2016. doi: 10.1007/978-3-319-45823-6_63. URL https://link.springer.com/chapter/10.1007/978-3-319-45823-6_63.

Dídac Rodríguez Arbonès, Nataliia Y. Sergiienko, Boyin Ding, Oswin Krause, Christian Igel, and Markus Wagner. Sparse Incomplete LU-Decomposition for Wave Farm Designs Under Realistic Conditions. In *Proceedings of the 15th International Conference on Parallel Problem Solving from Nature*, pages 512 – 524. Springer, September 2018.

Christopher M. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, 1995.

Trevor F. Cox and Michael A. A. Cox. *Multidimensional Scaling*. Chapman & Hall/CRC, 2000.

Olivier Du Merle, Pierre Hansen, Brigitte Jaumard, and Nenad Mladenovic. An Interior Point Algorithm for Minimum Sum-of-Squares Clustering. *SIAM Journal on Scientific Computing*, 21(4): 1485–1505, 2000. doi: 10.1137/S1064827597328327. URL https://epubs.siam.org/doi/10.1137/S1064827597328327.

Claude Maurice Fauquet, Dominique Desbois, Denis Fargette, and Georges Vidal. Classification of Furoviruses based upon the Amino Acid Composition of their Coat Proteins. In Joseph Ian Cooper and M. J. C. Asher, editors, *Viruses with Fungal Vector*, pages 19 – 36. Association of Applied Biologists, 1988.

Alan Julian Izenman. *Modern Multivariate Statistical Techniques – Regression, Classification and Manifold Learning*. Springer, 2008. doi: 10.1007/978-0-387-78189-1. URL https://www.springer.com/de/book/9780387781884.

Mehdi Neshat, Bradley Alexander, Markus Wagner, and Yuanzhong Xia. A Detailed Comparison of Meta-Heuristic Methods for Optimising Wave Energy Converter Placements. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 1318 –1325. ACM, July 2018. doi: 10.1145/3205455.3205492. URL https://dl.acm.org/citation.cfm?id=3205492.

Brian David Ripley. *Pattern Recognition and Neural Networks*. Cambridge University Press, 1996. doi: 10.1017/CBO9780511812651. URL http://www.stats.ox.ac.uk/~ripley/PRbook/.

John W. Sammon, Jr. A Nonlinear Mapping for Data Structure Analysis. *IEEE Transactions on Computers*, C-18(5):401–409, 1969.

Junhua Wu, Slava Shekh, Nataliia Y. Sergiienko, Benjamin S. Cazzolato, Boyin Ding, Frank Neumann, and Markus Wagner. Fast and Effective Optimisation of Arrays of Submerged Wave Energy Converters. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 1045 – 1052. ACM, July 2016. doi: 10.1145/2908812.2908844. URL https://dl.acm.org/citation.cfm?id=2908812.2908844.