



Lamarr Institute Report A Template

Pascal Welke 
Lamarr Institute
University of Bonn
Bonn, Germany

Christian Bauckhage 
Lamarr Institute
Fraunhofer IAIS
St. Augustin, Germany

ABSTRACT

The Lamarr Institute Report template defines a common look and feel for Lamarr written documentation, such as Lamarr Coding Nuggets. The source code of this document shows how to use the Lamarr Institute Report Template to produce nice documents.

1 INTRODUCTION

One particular use case for this template are *Lamarr Coding Nuggets*. Coding nuggets are currently (short) how-tos that show how to solve problems “by foot” in practice. The typical structure of such a paper consists of motivating a (machine learning) problem, showing how to formulate this problem in a particular way and then showing how to solve it practically using existing methods.

Examples (from good old ML2R times) include “Solving Linear Programming Problems” [2] and “Linear Programming for Robust Regression” [1]. In the latter example, e.g. we show how to fit a linear model to data by minimizing least absolute deviations, instead of the more common least squares. To this end, we formulate the optimization problem, transform it to show that it can be solved by linear programming and then solve it in [python](#) using methods from [numpy](#) and [scipy](#) alone.

2 TECHNICAL DETAILS

preamble.tex contains several imports and definitions. There is a part marked optional that you can change, remove or add to as you please. Please refrain from changing the part, however, that is marked as non-optional. Below we give a short introduction to

2.1 Colors

The preamble defines the Lamarr Institute corporate design colors that should be used throughout the document and within figures (cf. Fig. 1) to provide a common look and feel:

blue : [lmrrblu](#)
yellow : [lmrryel](#)
red : [lmrrred](#)
turquoise : [lmrrtrq](#)

2.2 Python Code and keywords

Python code with pretty printing is supported. Inline code looks like this:

```
vecWLAD = result.x[:m]
print('w0 =', vecWLAD[0])
print('w1 =', vecWLAD[1])
```

An example of a longer, floating python code environment can be found in Listing 1.



Figure 1: A figure with our corporate design colors

Listing 1: setting up a linear programming problem

```
1 matA = np.zeros((2*n, m+n))
2 matA[:n, :m] = +matF.T
3 matA[n, :m] = -matF.T
4 matA[:n, m:] = -np.eye(n)
5 matA[n, m:] = -np.eye(n)
6
7 vecB = np.zeros(2*n)
8 vecB[:n] = +vecY
9 vecB[n:] = -vecY
10
11 vecC = np.hstack((np.zeros(m), np.ones(n)))
```

ACKNOWLEDGMENTS

This material was produced within the [Lamarr-Institute for Machine Learning and Artificial Intelligence](#) which is funded by the Federal Ministry of Education and Research of Germany and the state of North-Rhine Westphalia (grant no. LAMARR22B). The authors gratefully acknowledge this support.

REFERENCES

- [1] P. Welke and C. Bauckhage. 2020. *ML2R Coding Nuggets: Linear Programming for Robust Regression*. Technical Report. MLAI, University of Bonn.
- [2] P. Welke and C. Bauckhage. 2020. *ML2R Coding Nuggets: Solving Linear Programming Problems*. Technical Report. MLAI, University of Bonn.