Some Title

Some subtitle

Daniel Runge Petersen, Nutsy Superman
Computer Science, xXx

Semester Project

Composed and typeset by the authors using the LaTeX Document Preparation System, based on the AAU report template by Petersen [1].

Electronics and IT
Aalborg University
http://www.aau.dk

AALBORG UNIVERSITY
STUDENT REPORT

Title:
Project Title

Theme:
Scientific Theme

Project Period:
Fall Semester 2022

Project Group:
xXx

Participant(s):
Daniel Runge Petersen
Nutsy Superman

Supervisor(s):
Supervisor 1
Supervisor 2

Copies: 1

Page Numbers: 11

Date of Completion:
September 26, 2022

Abstract:
This is the abstract.

Elektronik og IT
Aalborg Universitet
http://www.aau.dk

AALBORG UNIVERSITET
STUDENTERRAPPORT

Titel:
Rapportens titel

Tema:
Semestertema

Projektperiode:
Efterårssemestret 2010

Projektgruppe:
XXX

Deltager(e):
Forfatter 1
Forfatter 2
Forfatter 3

Vejleder(e):
Vejleder 1
Vejleder 2

Oplagstal: 1

Sidetal: 11

Afleveringsdato:
26. september 2022

Abstract:
Her er resuméet

Contents

Preface

This is the preface. You should put your signatures at the end of the preface.

Aalborg University, September 26, 2022

Author 1
<username@student.aau.dk>

Author 2
<username@student.aau.dk>

Author 3
<usernamestudent.aau.dk>

Author 4
<username@student.aau.dk>

Author 5
<username@student.aau.dk>

Author 6
<username@student.aau.dk>

Chapter 1

Introduction

Welcome to Aalborg University (AAU). Here we use LaTeX to typeset our high quality reports. Checkout the Appendix A for examples.

Chapter 2

Problem Statement

2.1 Audio

For models that predict what music is popular or what genre the music is we would like to see how big of an effect feature engineering has for the model. We would like to investigate which kind of dimensionality reduction works best considering both linear and nonlinear aproaches and what they contribute to in the model and when it is a better fit. The performance of these dimensionality reductions is evaluated based on how they affect the performance of the model and their visualisations.

2.2 Pokemon

For a model that clasifies Pokemon we would like to see how big of an effect feature engineering has for the model. We would also like to investigate which kind of dimensionality reduction works best and consider both linear and nonlinear aproaches and what they each contribute and when theyre correct to use. The performance of these aproaches might be evaluated based on their visualisations and how they affect model performance.

2.3 Match data

For models that predict the outcome of football matches we would like to see how big the effect of feature engineering has for the model. We would also like to investigate which kind of dimensionality reduction works best considering both linear and nonlinear aproaches and what they contribute to in the model and when it is a better fit. The performance of these dimensionality reductions is evaluated based on how they affect the performance of the model and their visualisations.

2.4 ??

The problem is clasification of Pokemon types. This will be done by using machine intelligence models to predict the type of a pokemon based on its colours. The 2d-images will be pre-proccesed to have the same size and clasified based on their RGB colours values, using both linear and nonlinear dimensionality reduction methods. The metrics of the evaluation will be speed as images contains large amount of data.

Clasification on pokemons, there are 3 different colours for each type of pokemon a combination of RGB

predict color in grey scale based on the color of the pokemon

Acronyms

AAU Aalborg University. 1

Bibliography

[1] Daniel Runge Petersen. *AAU-Dat templates*. URL: https://github.com/AAU-Dat/templates (visited on 08/17/2022).

Appendix A

Examples

When working in LaTeX we have basic text, such as this, and non-basic elements called 'floats'. They are called floats because they float about the page, trying to be as unobtrusive as possible. The common floats used by us Computer Science students are: Figures, Tables, Listings, and potentially Algorithms.

A.1 Figures

Figures include an image or a graphical frame of some sort.

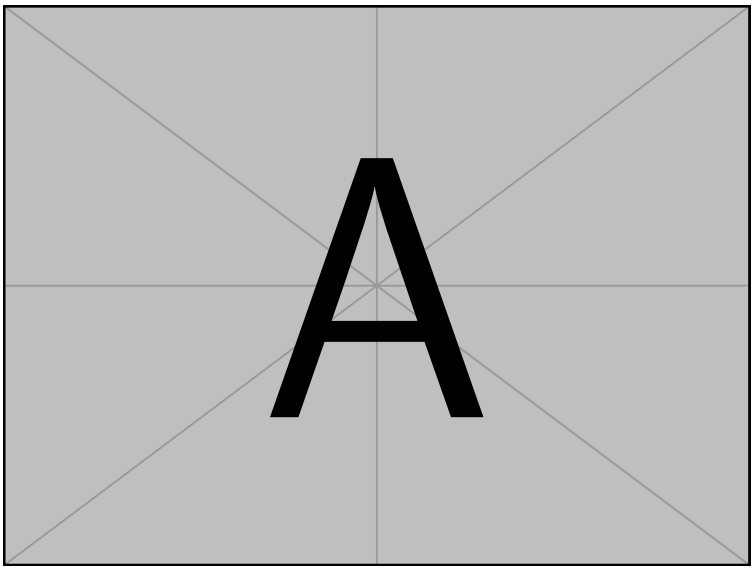

Figure A.1: An example of a figure float with width at 70% text width - **do not** use scale.

A.2 Tables

Tables are often more difficult than figures, but they can look gorgeous.

Features	Events	Threads	**Protothreads**
Control structures	no	yes	**yes**
Debug stack retained	no	yes	**yes**
Implicit locking	yes	no	**yes**
Preemption	no	yes	**no**
Automatic variables	no	yes	**no**

Table A.1: An example table. Do not use *vertical* (|) lines if you can avoid it.

A.3 Listings/Algorithms

We can use the listing float for both code and pseudocode, but in case you want to distinguish between them, the Algorithm environment is a good substitute for pseudocode.

```
#include <stdio.h>

int main() {
    printf("Hello World!");
    return 0;
}
```

Listing A.1: Example of C code with standard styling.

```
1  import numpy as np
2
3  def incmatrix(genl1,genl2):
4      m = len(genl1)
5      n = len(genl2)
6      M = None #to become the incidence matrix
7      VT = np.zeros((n*m,1), int)  #dummy variable
```

Listing A.2: Example of python code with custom styling from preamble and lines highlight.

For pseudocode, if you don't want to use the listing float, you can use the algorithm float instead.

INSERTION-SORT(A)

```
1  for j = 2 to A.length
2      key = A[j]
3      // Insert A[j] into the sorted sequence A[1..j-1].
4      i = j - 1
5      while i > 0 and A[i] > key
6          A[i+1] = A[i]
7          i = i - 1
8      A[i+1] = key
```

Algorithm A.1: Example of pseudocode (codebox) in an algorithm float.

A.4 Floats and content

Note that Figures use includegraphics, Tables use tabular, Listings use minted and Algorithms use codebox in the above example. This is NOT a hard and fast rule. You could have an Algorithm with an includegraphics screengrab, or a Listing with a tabular to create columns for code side by side comparisons.