UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

FACULTY OF ENGINEERING, BUILT ENVIRONMENT
AND INFORMATION TECHNOLOGY

DEPARTMENT OF ELECTRICAL, ELECTRONIC AND COMPUTER ENGINEERING
http://www.up.ac.za/eece

STUDY GUIDE (2021) FOR

EAI732: Intelligent Systems

Compiled by

Dr A de Freitas, Prof JP de Villiers, Mr H Grobler, Prof JP Jacobs and Prof HC Myburgh

19 March 2021

ISG@UP
INTELLIGENT SYSTEMS GROUP
http://isg.up.ac.za/

Contents

1 Introduction

The module EAI732 is the first and primary module in the postgraduate program of the Intelligent Systems Group (ISG) and lays the foundation for the diverse research focus areas of the group. The name of this module embodies the two elements fundamental to the research focus: intelligence and systems. The research focus of the ISG therefore lies in the theory and methods whereby various degrees of intelligence can be simulated in practical systems. Although intelligence can be applied in various real-world applications, and the ISG activities include such applications, the goal of the ISG is the development of new approaches to simulated intelligence. The assumption is that such new approaches will be tested in various applications, ranging from the obvious such as robots, to the less obvious such as data fusion, music and planning systems. The various applications and specific systems being developed therefore represent particular test cases of more general theory and methods being developed.

Before new approaches can be developed, or established approaches applied to new applications, the existing approaches to simulated intelligence must be studied and hence the need for this module. With roots in the field of cybernetics, simulated intelligence has been the subject of research since the early 1920s. The research gained significant momentum with creation of the discipline of Artificial Intelligence (AI) in 1956. Over time various sub-fields have been defined which focus on specific aspects of the larger problem of simulated intelligence. Of these, Computational Intelligence (CI), Machine Learning (ML) and Pattern Recognition (PR) are a few examples. The field of simulated intelligence is therefore a well established field and a large collection of techniques and methods have been developed.

The goal of the module EAI732 is to provide postgraduate engineering students with both theoretical and applied knowledge of the field of intelligent systems. The official scope of the module is defined as:

> This module provides the theoretical background necessary to understand, research and develop real-world software and hardware systems that incorporate and exhibit intelligent behaviour. The module incorporates advanced theory from fields such as Artificial Intelligence, Computational Intelligence, Machine Learning, Pattern Recognition and Signal Processing. Core topics of the module include: Bayesian Theory, Neural Networks, Kernel Methods, Graphic Models, and Numerical Bayesian Methods.

Due to the vast number of concepts and techniques which falls within the domain of intelligent systems, only selected topics can be covered in this module. The aim will be to explore theory and techniques that represent the dominant constructs applicable to intelligent systems. The majority of these techniques will derive from the sub-fields of Machine Learning and Pattern Recognition. Although the module contents is mathematical in nature, the target applications are real-world and practical. The assignments are structured to give the student practical experience with the techniques studied.

In conjunction with the EAI732 module, postgraduate students in the ISG will also complete the EAI733 Advanced Topics in Intelligent Systems module. The contents of the EAI733 module is custom designed for each student with focus on their intended Masters research. The module will typically incorporate advanced theory from fields such as: Digital Image Processing, Computer and Robotic Vision, Probabilistic Robotics, Data Fusion, etc. The EAI733 module therefore allows the student to obtain specialised background in particular research areas upon which their future Masters research will build.

2 Lecturer Information

Lecturer:	Dr A de Freitas BEng(Pret), BEngHons(Pret), MEng(Pret), PhD(Sheffield)
Office:	Engineering 1, Room 15-18 Main Campus University of Pretoria
Mail address:	Department of Electrical, Electronic and Computer Engineering University of Pretoria Pretoria, 0002
E-mail address:	allan.defreitas@up.ac.za
Consulting hours:	By appointment. The preferred method of contact is email.

Lecturer:	Prof Pieter de Villiers BEng(UP), MEng(UP), PhD(Cantab), Pr.Eng(ECSA)
Office:	Engineering 3, Room 7-54 Main Campus University of Pretoria
Mail address:	Department of Electrical, Electronic and Computer Engineering University of Pretoria Pretoria, 0002
E-mail address:	pieter.devilliers@up.ac.za
Consulting hours:	By appointment. The preferred method of contact is email.

Lecturer:	Mr. H. Grobler BEng(EE)(Stell), MScEng(EE)(Stell), BScHons(CS)(Pret), MSc(CS)(Pret)
Office:	Engineering 1, Room 13-17 Main Campus University of Pretoria
Mail address:	Department of Electrical, Electronic and Computer Engineering University of Pretoria Pretoria, 0002
E-mail address:	hans.grobler@up.ac.za
Consulting hours:	By appointment. The preferred method of contact is email.

Lecturer:	Prof JP Jacobs Pr.Eng, BEng(Pret), MEng(Pret), PhD(Pret), BMus(UNISA), BMus(Hons)(Pret), MMus(Pret), MMA(Yale), DMA(Yale)
Office:	Engineering 1, Room 13-15 Main Campus University of Pretoria
Mail address:	Department of Electrical, Electronic and Computer Engineering University of Pretoria Pretoria, 0002
E-mail address:	jpjacobs@up.ac.za
Consulting hours:	By appointment. The preferred method of contact is email.

Lecturer:	Prof HC Myburgh BEng(Pret), BEngHons(Pret), MEng(Pret), PhD(Pret)
Office:	Engineering 1, Room 15-5 Main Campus University of Pretoria
Mail address:	Department of Electrical, Electronic and Computer Engineering University of Pretoria Pretoria, 0002
E-mail address:	herman.myburgh@up.ac.za
Consulting hours:	By appointment. The preferred method of contact is email.

3 Prerequisites

Whilst there are no ***official*** prerequisites for EAI732, there are a number of implied prerequisites. A student who does not meet the majority of these requirements will find the module challenging and takes the module at their own risk. It is assumed that students will perform any necessary self-study to compensate for a lack with regards to these implied prerequisites. Students taking this module are assumed to have an interest in and commitment to understanding the concepts in depth. It is also important that students take an integrated approach and actively seek connections between the various topics and their existing knowledge.

- **Artificial Intelligence (AI)**: It is assumed that the student has completed an introductory course on AI concepts, such as the undergraduate module "EAI320 Intelligent Systems". In particular, basic knowledge of Bayesian approaches will be assumed.
- **Strong mathematical background**, with particularly emphasis on: linear algebra, probability theory and statistics.
- **Programming proficiency**: Students will be expected to implement algorithms and should therefore be proficient programmers in Python (preferred), MATLAB / Octave or C/C++. If the student is not proficient in one of these languages, the student will need to budget additional time for assignment implementation. This will most likely result in the workload exceeding the nominal 20 hours per week.
- **Research mindset and critical thinking**: It is assumed that students approach this module with a willingness and eagerness to learn and perform critical thinking. Numerous papers will be prescribed and students are expected to perform critical analysis of such papers to extract the essentials.

Note that from 2021 the EAI732 module will only available to students of the UP Department of Electrical, Electronic and Computer Engineering, or students with an ISG staff member as postgraduate supervisor or co-supervisor.

4 Prescribed Text

The textbook selected for this module is:

- **Textbook**: Christopher M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 1st ed. 2006. Corr. 2nd printing 2011 edition (October 1, 2007), ISBN: 9780387310732. The home page of the book, which contains errata and other resources which should be consulted, is https://www.microsoft.com/en-us/research/people/cmbishop/#!prml-book.

The textbook has a strong mathematical and statistical flavour, and will typically require some effort to master the details presented. The textbook is also notable for its dense nature, with nearly every statement linked to fundamental and important concepts. As a result the true value of the material can typically only be appreciated after multiple readings. Although the mathematical and statistical details are important, students should take care not to get lost in these details. When first studying a theme, focus on the concepts rather than the details.

The textbook covers a wide variety of topics and, for the purpose of this module, only selected chapters will be studied and typically only to a limited level of detail. Depending on the intended Masters topic of the student, the student may revisit the studied chapters or other chapters in more detail as part of

the module EAI733. More detailed information on the work covered in this module and supplementary references will be given in work packages as described in section 11.

5 Course Objectives and Study Themes

The EAI732 module focuses on selected constructs that form the basis of many systems that incorporate some degree of intelligence or a high degree of adaptability. It must be understood that these selected constructs are but a small subset of a much larger set of available techniques. The constructs that will be studied have been selected based on their inherent power and position in the hierarchy of the available constructs.

The module content is divided into a number of themes corresponding approximately to the contents of the prescribed text. The schedule according to which the themes will be studied are provided in section 12.

- **THEME 1**: ***Introduction and Overview of Concepts***: Revision of Probability and Bayesian Theory. Models for Regression and Classification. Model Parameter Estimation.
  - ***Introduction* – Bishop Chapter 1**: Introduction to the concepts of Regression and Classification. Revision of Probability and Bayesian Theory. Introduction to Decision Theory.
  - ***Probability Distributions* – Bishop Chapter 2**: Binary Variables and Multinomial Variables. The Gaussian Distribution. The Exponential Family. Non-parametric Methods.
  - ***Linear Models for Regression & Classification* – Bishop Chapter 3 & 4**: Basis Functions. Maximum Likelihood and Least Squares. Bayesian Parameter Estimation. Discriminant Functions.
  - ***Mixture Models and EM* – Bishop Chapter 9**: k-Means Clustering. Mixtures of Gaussians. Expectation Maximisation.
- **THEME 2**: ***Neural Networks and Optimisation***: Feed Forward Neural Networks. Network and Model Optimisation. Heuristic Algorithms.
  - ***Neural Networks* – Bishop Chapter 5**: Network architectures. Network Training. Backpropagation algorithm. Regularization. Practical techniques for backpropagation.
  - ***Optimisation and Heuristic Algorithms* – Supplementary Material**: Alternative training / optimization algorithms. Evolutionary and swarm approaches to optimization.
- **THEME 3**: ***Kernel and Graphical Methods***: The Kernel Trick and Kernel Substitution. Gaussian Processes. Support Vector Machines. Graphical Models.
  - ***Kernel Methods* – Bishop Chapter 6**: The Kernel Trick. Constructing Kernels. Radial Basis Functions. Gaussian Processes.
  - ***Support Vector Machines* – Bishop Chapter 7**: Maximum Margin Classifiers and Support Vector Machines. Relevance Vector Machines.
  - ***Graphics Methods* – Bishop Chapter 8**: Bayesian Networks. Inference in Graphical Models.
- **THEME 4**: ***Sequential Methods***: Sampling Methods. Sequential Models and Methods.
  - ***Sampling Methods* – Bishop Chapter 11**: Sampling Algorithms. Markov Chain Monte Carlo. Gibbs Sampling.
  - ***Sequential Data* – Bishop Chapter 13**: Markov Models. Hidden Markov Models.

Upon completion of this course, the student should:

- understand the **concepts** and **application areas** of selected machine learning constructs,
- understand the **algorithms** used to implement these constructs in practical systems,
- be able to **design** and **implement** algorithms to solve sample machine learning and pattern recognition problems, and
- be ready to **solve a real-world problems** by designing an appropriate system and implementing the theory and algorithms studied in this course, as will be required for the Masters.

6 Requirements and Methodology

In this module an applied approach is taken. Once the student has mastered the theoretical fundamentals, the student will be expected to apply the techniques and methods to solve all aspects of the various assignment problems. Where possible, students should strive to implement the algorithms from first principles, however this is not a requirement. In some cases the complexity of the methods may prohibit implementation from first principles and existing implementations may be used. In either case, students are encouraged to compare their implementation with that of existing implementations. If such comparisons, together with a critical analysis, are included in the assignment submission the student may be awarded bonus marks.

Students taking this course are assumed to have knowledge of basic linear algebra and probability theory. It is further assumed that students still have their undergraduate mathematics handbooks and will revise the necessary theory as required. Where appropriate references will be supplied to supplement and extend the student's current knowledge. Note that such revision is not seen as part of this course and therefore not included in the time / load calculations.

As honours level study is seen as preparation for a Masters degree, students need to transition from a structured program (*i.e.* undergraduate level study) to a more research orientated approach. The methodology followed will therefore be predominately that of guided self-study. The textbook is considered to provide a representative view of the subject field for the purposes of this course. Students will be required to study the chapters of the book according to the schedule given in section 12 below. Assignments will allow both the lecturers and the students to evaluate progress.

The student will not be expected to know the material off by heart but must have a working understanding of the material and must be able to apply the knowledge to practical implementations. It must however be stressed that the student is expected to have complete understanding of the theory and ***must be able to derive all mathematical aspects studied unless otherwise stated***.

Although the prescribed textbook is considered the best reference within the context of this module, at times supplementary material will be prescribed. In most cases these will be journal papers, conference papers, tutorials, in some cases video lectures, and this material will be made available under the class directory (see section 11). Students are expected to study the supplementary material and generally the material will provide information that will assist with the assignments. However, the primary aim of the supplementary material is to broaden understanding and provide additional perspective. A secondary aim of the supplementary material is to serve as starting point for further study during Masters.

The study of substantial amounts of additional material is something that is particularly different from undergraduate courses. If one is not used to this it can appear daunting. Generally as you progress beyond a first degree the amount of background reading required increases. Due to the volume of material one clearly cannot study everything in detail. Learn to skim / speed read material to detect essential concepts. The ability to do so comes with practise. The goal is to expand understanding and to be aware of the contents of the material so that one can locate relevant information for a particular assignment / problem. Similarly, it is expected that for each assignment the additional references be studied to extract material relevant to the assignment. In summary, students are expected to do their

own filtering of material.

A mailing list, with address "`eai732@up.ac.za`", will be used for general communication with EAI732 students, to point out aspects of particular importance, and to provide assignment hints. Students can and should also use this email address for asking questions (that either the lecturers or fellow students can answer). Students are also welcome to use the mailing list to form and coordinate study groups. Mail from the mailing list will contain a "`[eai732]`" prefix, automatically added by the mailing list system. Questions posed to the lecturers via other media will also be answered via the mailing list to ensure that all students receive the same answers. All correspondence directly with the lecturers must be identified with a "`{eai732}`" prefix in the subject line of the mail. Please remember to always include your student number in email directly to the lecturers.

7 Recommended Approach

Due to the methodology followed for the module care must be taken to avoid spending time on less important aspects or trying to fix badly developed implementations. Most students who reach postgraduate level have established an effective study / research methodology for themselves. There are however students who have not yet developed such a methodology and it is recommended that they approach the study themes and assignments as follows:

1. Begin by creating a new report file using the assignment report template. It is assumed that each student will create such a template during the first week of the semester, based on the report guidelines provided in Appendix A.

2. Scan through all of the designated chapters in the textbook associated with the study theme to obtain a global view of the theme.

3. Read through the entire work package and assignment description for the study theme to determine what aspects are considered important and what theory will be needed to complete the assignment. In many cases the work packages also include hints on the assignments that can save considerable time and effort.

4. Now study all of the material in the textbook relating to the study theme in more detail. Particular emphasis should be placed on the aspects that are the focus of the assignment.

5. Whilst studying the material, begin writing the sections in the report that entail the literature review, theoretical analysis (*e.g.*, background / fundamental theory, possible approaches, motivations of assumptions / approximations, derivation of all required equations, *etc.*) and the algorithmic development (*i.e.* pseudo-code with final equations inline or references to derived equations). For certain material, such as plots and diagrams, use placeholders and revisit these sections when finalising the document. At this stage aspects of the material in the textbook may be unclear or the details of certain aspects may be missing.

6. Consult the additional references provided in the work package document for the various sections. Begin by scanning through the material to obtain an overview and to determine how the references contribute to the subject material. Study particular supplementary references in more detail as their importance becomes clear. Perform additional research using, for example, Google Scholar and the UP Library Database of Electronic Resources (e-Journals).

7. The above three steps should be repeated until the theoretical analysis and algorithmic development have been completed and a clear understanding of both has been established. Remember to

keep the objectives of the assignment in mind to help remain focused. At this stage the bulk of the study material in the textbook should have been mastered. If not, pose any questions to the mailing list.

8. Where feasible, experiment with existing implementations in order to further increase understanding of the assignment topic and the expected outcomes.

9. Based on the algorithm developed / extracted, create an implementation and perform the necessary tests on suitable test data to verify that the implementation is correct. If possible, compare the implementation to existing implementations explored in the previous item (this will allow the student to obtain a direct indication that they are on the right track). In the report provide a brief description of the implementation and the basic tests done to validate the implementation.

10. If applicable, study the dataset specified to be used to provide a comprehensive evaluation of the implementation. Provide an overview of the dataset in the report, remembering to include a reference for the dataset.

11. Develop and include in the report an experimental plan for the comprehensive evaluation of the implementation, as appropriate. It is also important to clearly define the criteria that will be used to evaluate the results of the experimental evaluation. Many problem domains have standard protocols or criteria that may be applicable and these should be researched and used as appropriate. For example, in the case of image processing related problems, a wide selection of image quality metrics have been defined that can be used to evaluate the relative effectiveness of an algorithm and / or implementation. Document the experimental plan and evaluation criteria in the report taking care to motivate all aspects.

12. Perform the experimental evaluation and provide a suitably concise summary of the results in the report, together with a discussion of the results.

13. At this stage the implementation and the body of the report has been completed. Perform any necessary editing to finalise the report body.

14. Now write the introduction and conclusion of the report based on everything that has been learned during the execution of the assignment. Make sure that the introduction explains the purpose of the assignment within the larger subject matter and that the conclusion highlights to what degree this has been achieved.

15. The final writing task is that of the abstract. This is often one of the most difficult writing tasks and can require a surprising amount of time. Writing good abstracts is somewhat of an art form and often one must rewrite the abstract 2 or 3 times to obtain the desired result. Practise is therefore important and is one of the reasons for the abstract requirement in the reports.

16. Include and finalise any graphical and tabular material required (*i.e.* where you previously left placeholders).

17. Complete the report by proofreading the document, checking for any errors in content, style, language, structure and layout.

To reiterate, the assumption is that the student will begin writing the report from the moment work begins on the assignment. Only once the necessary derivations and pseudo-code has been documented in the report should actual implementation begin. Attempting to create an implementation without a correct algorithm as basis will invariably lead to an incorrect implementation. Unnecessary time will

then be required to fix the implementation and there will most likely not be sufficient time to complete the assignment. The report should therefore reflect the systematic investigation of the study theme and assignment problem. The systematic investigation reflected in report is used to evaluate the degree to which a student has mastered the study themes.

Similarly, before any experimentation / simulations are done, an experimental plan must be developed and documented in the report. Note however that many of the experiments can take considerable time to complete and careful planning is therefore required. It is also important to perform as much as possible of the experimental evaluation and results processing in parallel.

In the report you should describe the algorithms implemented. The "algorithm" environment in LaTeX is ideal for this. Wherever you make use of an equation in your algorithm, ensure that there is a corresponding derivation / explanation of the equation. Note that this aspect is considered to be a crucial aspect of the learning process. ***You must fully derive, from first principles, all equations used in the implementation(s). The derivations must be given in the report, failure to do so will result in reduced marks unless you motivate fully why such derivation is not applicable / feasible.***

8 Assessment

8.1 Grading Policy

Please see the "Degree Requirements" section of the EECE Postgraduate web pages for general information regarding module load and mark requirements. In particular, the time-load of a 32-credit module is taken to be a nominal 320 hours (for the average student), excluding revision time. This translates to approximately **20 effective hours per week**. Unfortunately most students either find that their mathematical background is lacking and / or their programming skills are not sufficient, and 30 – 40 hours per week are typical in such cases. The final mark for EAI732 will be based on evaluation during the semester in the form of assignments. See section 11 for information on the submission of the assignments.

The semester mark and final mark are made up as follows:

	effective hours	**% of semester mark**	**% of final mark**
Theme 1 (+ Assignment 1)	74	25%	25%
Theme 2 (+ Assignment 2)	74	25%	25%
Theme 3 (+ Assignment 3)	74	25%	25%
Theme 4 (+ Assignment 4)	74	25%	25%
Contact Time	24		
TOTAL:	320	100%	100%

8.2 Late Assignment Policy

The module schedule given in section 12 includes the *due date* of each assignment and the submission of the assignment is due at 23:59 on that day. On the AMS, the *due date* is indicated as the "Assessment Date".

Recognizing that students may face unusual circumstances, each student will have a quota of seven free late (calendar) days to use as he / she sees fit. Once these late "grace" days are exhausted, any assignment submitted late will be penalized 20% per day late (that is an actual 20% penalty, **not** 20% of mark obtained). Each 24 hours or part thereof that an assignment is late uses up one full late day.

Apart from the late "grace" day quota, there is also a per assignment four late day limit after the

due date. On the AMS this is represented as the submission "Upload Deadline". The AMS will automatically apply a late penalty as determined by the standard AMS late submission policy given at bottom the AMS submission page. If your grace days have been exhausted, and you also submit after the assessment submission deadline, *both* penalties will be applied.

8.3 Semester Tests and Examination

No semester tests or final exam are applicable for EAI732. The final year mark obtained will be based on the assignment marks.

9 Class Attendance

At postgraduate level there are no classes as is typical at undergraduate level. Most of the postgraduate subjects make use of a block week format where one spends a week at a time attending lectures. Amongst other, the motivation for this format is that many students study part-time and may not be able to attend regular classes.

This format will also be used for the current module. There will be postgraduate lectures on the Monday and Tuesday of the block weeks. Typically the lectures start at about 08:30 and continue until about 13:00, depending on questions addressed / discussions. The agenda for each block week will be communicated approximately one week before the block week.

Interaction with students occurs primarily through the mailing list. This model has proven to be more effective than the other models. During the block weeks, as well as any other *arranged* time, the lecturers will be available to answer any queries not yet answer via the mailing list.

Students may be required to demonstrate and answer questions regarding any previously completed assignment. This will typically be necessary if a particular student's progress is unsatisfactory. Such students will be notified via email with the venue and time.

In the event that fact-to-face / campus classes are not possible, these will be replaced with videos, video conference or narrated slides as determined by the lecturer responsible for the particular class.

10 Ethics

Students are encouraged to discuss course work with each other, the mailing list being the recommended media. However, each student should hand in his/her own work for assignments. Plagiarism in the submitted report, including copying the work of another student and copying from the internet, is absolutely unacceptable. With respect to the submitted code, and as indicated in section 6, existing implementations may be used. However, all code fragments imported from other sources should be clearly marked as such, ideally with the source URL included in a comment before the reused code fragment.

At postgraduate level, the official policy of the University of Pretoria is that ANY level of plagiarism immediately results in a disciplinary hearing being called and will seriously jeopardise the academic future of the student. In addition, please note the following (provided by the University's Legal Services Department):

> Under the definitions of misconduct a student is guilty of misconduct if he/she is guilty of any conduct that infringes copyright or any other form of the law of immaterial property and such conduct proves to be detrimental to the University.
>
> The inclusion of the work of other authors (literacy works) in dissertations and theses has to be done in accordance with the provisions of the Copyright Act, 98 of 1978. This Act states that

> the copyright in a literary work (also if made available electronically) shall not be infringed by a short quotation there from provided that the source shall be mentioned as well as the name of the author. Non-compliance with these provisions will therefore not only be a contravention of the Rules of the University, but also a crime in terms of the South African Law.

All reports submitted by the student must include a signed declaration of originality, see Appendix B. Any assignment that does not include the scanned statement with hand written signature will be awarded a mark of 0. Note that assignments are tested for plagiarism via Turnitin, as per the University Plagiarism Prevention Policy.

11 Work Packages and Assignments

A number of assignments are due for this course. The work covered in the module is divided into sections that correspond with the study themes defined in section 5. Whereas this guide provides information on the administrative aspects of the module, the details of the work covered in a specific study theme is described in a document (essentially volume II of the study guide) referred to as a "Work Package". These work packages are available in the file "`eai732_wps.pdf`" in the top of the class directory (see next paragraph). The work packages contain references with active links to supplementary material. The material will automatically be opened if the material is available. For this to work the same relative directory layout is required, *i.e.* in the same subdirectories as indicated in the work package. The specifications for the assignment related to a specific section of work will be included at end of each work package.

11.1 Accessing Supplementary Material

Once registrations for the module have been completed, all students will receive a Google Drive notification with link to the read-only share containing the supplementary material. Note that the complete class directory is large and it is recommended students visit campus to download the entire contents (if so desired).

11.2 Assignments Topics and Execution

Programming projects must be implemented in Python, Octave/MATLAB or C/C++ unless otherwise arranged. Java is generally not recommended as the garbage collector tends to cause significant performance issues with these data intensive types of applications (unless steps are taken to tune the Java VM).

Some of the assignments may require considerable computational time to "train" or optimize the implemented models. Additionally a large number of repetitions are typically required to produce statistically valid results.

The datasets required by some of the assignments will be made available in the assignment directory. The lecturers may test your code during evaluation and your code must therefore execute using standard environments (*e.g.*, Anaconda for Python, MATLAB or Octave, *etc.*).

The following is a list of assignment topics. The due dates for the assignments are included in the schedule (section 12).

- **Assignment 1**: Probability Theory. Bayesian Theory, Inference and Parameter Estimation.
- **Assignment 2**: Neural Networks and Optimisation.
- **Assignment 3**: Kernel and Graphical Methods.
- **Assignment 4**: Sequential Methods.

11.3 Submission of Assignments

The Judicator Assessment Management System (AMS) will be used for the submission and management of assessments: https://ams.up.ac.za. If you have not previously used the AMS, you need to set a new password using the "Forgot or change password" link on the login page. Note that this will only be possible once the official class list has been loaded.

To submit an assignment, select the module and assignment from the list. If submissions are enabled, the "Submit Upload" button will be active and should be used to upload the assignment files.

The marking scheme used for each assignment will be visible on the AMS when the assessment is opened for submissions.

The file names for the submitted files should be of the following format:

`EAI732A<X>_<studentnumber>_<Surname>_<Initials>_report.pdf`

`EAI732A<X>_<studentnumber>_<Surname>_<Initials>_code.zip`

where <X> is the assignment number, `<Surname>` is your surname and `<Initials>` is your initials. If you have a multi-word surname, please capitalise (proper case) each word and concatenate the result (*e.g.*, the report of assignment 1 for "Mr PLR van der Merwe" should be saved in a file with name similar to "`EAI732A1_12345678_VanDerMerwe_PLR_report.pdf`"). Note that if the above format is not followed your submission may be ignored.

The implementation files that are part of the ZIP file should be placed (in the ZIP file) under a subdirectory named "`code`". A plain text file "`code/README.txt`" should also be included which contains instructions on how the implementation should be built or necessary files generated. The file should also include instructions on how to reproduce the results as described in the report.

11.4 Evaluation of Assignments and Feedback

After the cut-off deadline for an assignment the marking process begins. Once an assignment has been marked, an evaluation of the assignment will be made available addressing any issues encountered. The target for the completion of assignment evaluation is approximately 1.5 weeks after the final submission has been received. Unfortunately due to the lecturers work-load it may not always be possible to achieve this target.

Once assignment evaluation has been completed, a rubric will be visible on the AMS which will provide an indication of how marks were awarded. For a time you will also be allowed to register queries regarding the awarded marks, by clicking on the particular mark.

Unfortunately time available for evaluations is restricted and this limits the depth to which evaluation can realistic be done, and hence the extent of comments added to the rubric. Therefore, please see the comments as hints at the most significant issues in the submission that could be identified in a limited evaluation time-frame.

> ***Assignments may be evaluated by attempting the execution of the code submitted. Code that does not work will result in penalties in the form of reduced marks.***

Note that if during the evaluation there is uncertainty with regard to your assignment (report or code), you may be asked to demonstrate / explain your code or report. The venue and time will be supplied. Students must note that the instructions given in the assignments are to be taken literally, and meticulously adhered to. Students that prefer not to follow the instructions given do so at risk of significantly reduced marks.

The final module marks will be made available once the external moderator has completed the moderation process. The target date for the availability of the final marks (on the UP Student Administration

System), is the week before the start of the second semester, depending on the moderation time required by the moderator.

12 Schedule

Week	Date	Theme	Required reading / Assignment due date
1	15 Mar – 21 Mar	1	Introduction - Bishop Chapter 1.
2	22 Mar – 28 Mar	1	Probability Distributions - Bishop Chapter 2.
3	29 Mar – 4 Apr	1	Regression - Bishop Chapter 3.
4	5 Apr – 11 Apr	1	Classification - Bishop Chapter 4.
5	12 Apr – 18 Apr	1	Mixture Models and EM - Bishop Chapter 9.
6	19 Apr – 25 Apr	2	**Postgraduate Block Week 1**. Neural Networks - Bishop Chapter 5. *Assignment 1 (due 19 April).*
7	26 Apr – 2 May	2	
8	3 May – 9 May	2	Optimisation and Heuristic Algorithms.
9	10 May – 16 May	2	***Recess***
10	17 May – 23 May	3	Kernel Methods - Bishop Chapter 6. *Assignment 2 (due 17 May).*
11	24 May – 30 May	3	Support Vector Machines - Bishop Chapter 7.
12	31 May – 6 Jun	3	Graphical Methods - Bishop Chapter 8.
13	7 Jun – 13 Jun	3	**Postgraduate Block Week 2**.
14	14 Jun – 20 Jun	4	Sampling Methods - Bishop Chapter 11. *Assignment 3 (due 14 June).*
15	21 Jun – 27 Jun	4	
16	28 Jun – 4 Jul	4	Sequential Data - Bishop Chapter 13.
17	5 Jul – 11 Jul	4	
18	12 Jul – 18 Jul		*Assignment 4 (due 12 July).*

Appendix

A Assignment Guidelines

A.1 General

The two main deliverables for assignments are a typed report (in PDF format) and source code (MATLAB, C, *etc.* and related support files). The report should be of an acceptable standard. In the case of a compiled implementation language, please **do not** include compiled binaries or executables. They will be build during evaluation and hence you must included all components required to build your implementation. In general, do not include any file that is generated apart from the report PDF.

For some assignments datasets are to be processed into intermediate files, typically by means of scripts. Remember to include all scripts to generate the needed files from source or original files, which are assumed to be in the current directory (symbolic links will be created as needed). Similar to the generated and intermediate files, the original datasets used should not be included in the submission ZIP file. The references section should contain a reference to the original dataset source.

> ***NOTE:*** *Please do NOT set the title in the properties field of the PDF document. By default PDF readers display the file name of the PDF document in the title bar of the PDF reader window. Conversely, if a title is set it will be displayed in the title bar of the window. This makes the marking a batch of submissions significantly more time consuming as it becomes difficult to keep track of the files being marked when a number of PDF documents are open simultaneously.*

A.2 LaTeX

It is highly recommended that the reports be written using LaTeX and postgraduate students of the ISG should become proficient with LaTeX as soon as possible. LaTeX produces documents of a much higher standard than typical word processors, in particular with regards to mathematical typesetting. Combined with the powerful table of contents and index generation system, as well as the automated bibliography management, alternatives are downright painful in comparison. Once a nominal proficiency with LaTeX has been attained, the time required to complete the reports will typically be less than if they were done in, for example, Word. Another advantage of LaTeX is that, due to its text based nature, one can create scripts to automatically generate content such as tables and plots directly from an implementation.

For an overview of the advantages of LaTeX, have a look at the The Beauty of LaTeX. Another aspect is the apparent non-deterministic behaviour of Microsoft Word. LaTeX documents written decades ago produce *exactly* the same output when processed with the latest version of LaTeX. This level of consistency is unheard of with Microsoft Word. On complex documents Microsoft Word tends to get confused at some point, and the result is unwanted effects in the document. It usually requires the document to be reassembled in order to correct these types of problems.

The following book will greatly assist the use of LaTeX and is a good investment.

- Frank Mittelbach, Michel Goossens, Johannes Braams, David Carlisle and Chris Rowley, *The LaTeX Companion*, Pearson Education Inc., 2nd edition, 2004, ISBN:0201362996.

Various other materials relating to LaTeX can be found in the "`latex`" subdirectory of the class directory.

A.3 Mark Allocation

The marks allocated for each assignment will be proportioned approximately as follows:

- Content: 90%
- Presentation 10%

By presentation, the following is implied: layout, logic flow, consistency, use of language, spelling, punctuation, referencing, *etc.* The composition of the mark for the report content components will be supplied with each assignment. As a general guideline, each report component will be evaluated based on a template mark out of 5 as follows (given the scope and target outcomes of the component):

- **0 – No Attempt** (*essentially no attempt made*)
  There was nothing useful to evaluate and no marks could therefore be awarded.
- **1 – Severely Deficient** (*attempt made but mostly incorrect*)
  Methodology, answers, results and/or implementations are mostly incorrect, superficial and have critical errors or problems. The answer shows little or no understanding and depth of the material. Presentation has fundamental problems in terms of structure, number of errors and technical quality.
- **2 – Unsatisfactory** (*attempt made but contains many errors, minimal level of description*)
  Methodology, answers, results and/or implementations are basically correct but have major errors or problems, or have limited depth. The answer shows a basic understanding of the material but limited depth. Presentation meets minimum standards but has significant problems in terms of structure, errors and technical quality.
- **3 – Acceptable** (*basic attempt with some errors, basic level of description*)
  Methodology, answers, results and/or implementations are mainly correct but have moderate errors and detail, or are not optimal. The answer shows moderate depth and understanding of the material. Presentation is generally good but has minor problems in terms of structure, errors and technical quality.
- **4 – Good** (*complete and essentially correct attempt together with sufficient level of description*)
  Methodology, answers, results and/or implementations are mainly correct and detailed but have minor errors, or are not optimal. The answer shows a very good depth and understanding of the material. Presentation is mostly well-structured and error free, and of good technical quality.
- **5 – Excellent** (*comprehensive and detailed submission*)
  Answers and results are correct, detailed and near-optimal, as is the approach and methodology used to obtain them. The answer shows an excellent, deep and synthesised understanding of the material. A degree of critical analysis is evident and alternatives have been explored. Additional research has been done (as appropriate) and references have been provided. The implementations are flexible and modular facilitating application on new problems. The material is well-structured, error free and of high technical quality.

The mark is estimated based on the same considerations applied in 4'th year project and MEng dissertation evaluations. As certain components are more complex than others, the total mark for the component may be a scaled version of the template mark, plus an additional factor to allow for finer mark grain. Components that contain notable errors will receive a template mark below 3 and as low as 1 depending on the severity of the errors.

A basic correct assignment, but with minimal motivation and methodology, will most likely obtain a mark of approximately 60 – 65%. If the assignment is complete and professional, and a degree of critical analysis is apparent (which indicates deeper understanding), a mark in the region of 80 – 85% can be obtained. Typically this will include a discussion of problems encountered, an analysis of potential solutions and how these problems were eventually solved. To obtain a mark higher that 85%, significant effort and/or understanding outside the exact specification will be required. This will typically also be associated with extra features and functionality in the implementation, as well as additional references indicating background research.

These comments describe the estimated effect of what is a composite marking process. The evaluation process has been so designed that a mark below 70% provides an indication that the student is not ready for postgraduate studies and should therefore reconsider attempting further studies.

A.4 Evaluation Methodology

The first aspect that is checked is the declaration of originality as the University considers plagiarism a very serious issue. Failure to include an acceptable statement will result in the submission being ignored. A zero mark will be awarded in such cases.

The assignments are evaluated at the component level in order to ensure unbiased assessment. A particular component is assessed for all students, before proceeding with a subsequent component. This approach, amongst other, allows the calibration of the evaluation procedure and allows the detection of interpretation problems with the assignment specifications. Due to the component level evaluation, students are requested to follow the approximate structure indicated in the mark break-down in order to simplify the marking process. It is however not required, nor desired, that the same headings as the marking sheet be used.

The "presentation" related components are typically assessed first, after which aspects such as layout, language and spelling are ignored for the "content" related components. It has been stated before: all submitted implementations are evaluated by studying the code and attempting to compile / synthesize and execute / simulate. In many cases the code is modified to test aspects and to display the results of certain operations. *If the code is difficult to study and / or execute, the attained mark will invariably be low.*

A.5 Technical Report Writing and Required Layout

Before any type of writing commences it is essential to define the target audience. The nature and level of education of the audience determines the optimal approach to be followed. For the reports to be submitted in this module the target audience is the module lecturers and the external examiner(s). The assumption is therefore that the target audience is already very well informed regarding the subject matter. The target audience therefore will not usually read the report in order to extract knowledge. Instead, the readers will look for evidence that the student has understood the material, can execute a systematic scientific / engineering study and can properly present the results.

The report should not only convey information clearly and coherently, but should also explicitly describe how and why the investigation was performed. The experimental methodology is of particular importance. The report is therefore the main mechanism used to evaluate the degree to which the student has mastered the study theme material (*i.e.* the thought processes that led to the implementation). The implementation is considered the final confirmation, the one without the other indicates only partial understanding.

The required format for the reports is approximately as follows:

1. **Title Page**: The essential information that must be present on the title page include: the university name, the faculty, the department, the course code and name, your name and student number, the number and title of the assignment, and the date of submission. The title of a report should be of the form "Technical Report Assignment X: Y", where X is the assignment number. Y is the designated title as given in the corresponding assignment specification (in parenthesis) and may be typeset across two lines. The title page of the study guide gives an example of the desired format.

2. **Declaration of Originality**: See Appendix B.

3. **Abstract and Keywords**: The report must include a page which contains the report title, your full name and affiliation, the date, an abstract and a list of specific keywords. The abstract should summarise the research that has been done and what the most significant outcomes were. The abstract should not be a description of the report, nor of the general research topic but should focus primarily on the contribution of research performed. An abstract must be self-contained and not refer to the rest of the document. The abstract is written assuming that the reader understands all relevant theory and is familiar with the types of experimental methods used. The abstract should be 150 – 220 words. The list of keywords must be as specific and unique to the research topic and report as possible given that they are used for indexing. For the same reason the abstract should also be as specific as possible and generic statements must be avoided.

4. **Contents**: The report must include a table of contents, with up to *two* levels of detail.

5. **Introduction**: The introduction of a technical report identifies the research topic, the purpose / objectives / goals, and the plan of development of the report. The research topic is the "what", the purpose is the "why", and the plan is the "how". Together these acquaint the reader with the context and overall execution of the assignment and this section will be evaluated based on how effectively this is achieved. The introduction should be 1 to 2 pages written in the students own words (*i.e.* do not just list the objectives in point form). Note that a comprehensive literature review is not required in this section. The necessary literature review should be integrated into the report body as part of the theoretical development sections.

6. **Report Body**: These sections should correspond to the report components designated in the mark breakdown.

7. **Conclusion**: The conclusion should summarise what was achieved (*i.e.* how the assignment goals were met).

8. **References**: A list of references in IEEE Transactions format is compulsory (see Presentation section below). Note that with only few exceptions, such as reference to a dataset, you should only include a reference to a paper that you have actually read.

9. **Appendix**: These contain material that is too detailed to include in the main report, such as detailed data of experimental analysis, if appropriate.

A.6 Content

In the report you must show clearly that you understand:

- the objectives / goals of the assignment
- the necessary theory, with particular emphasis on the mathematical aspects and context relative to existing / previous work

- how to apply the theory to create an implementation
- how to perform empirical and/or simulation based evaluation of the implementation
- the interpretation of the results

If the report does not contain a sufficient level of detail, written in your own words, the items above cannot be evaluated and the result will be a lower mark. For the evaluation of the implementation you must include sufficient, step-by-step information on how to simulate / execute your implementation. If you did not do so, or your implementation cannot be executed, it is assumed that your results cannot be reproduced and are considered invalid.

The latter aspect mentioned relates to the so called principle of *reproducible research* (RR), which is a fundamental element of the scientific method. Results and conclusions based on research that cannot be independently reproduced are considered invalid and cannot contribute to the existing body of knowledge. For this module adherence to the principle of reproducible research is considered essential. For an overview please read the following article:

- Patrick Vandewalle, Jelena Kovacevic and Martin Vetterli, "Reproducible Research in Signal Processing – What, why, and how," IEEE Signal Processing Magazine, Volume 26, Number 3, pp. 37–47, 2009.

A.7 Presentation

The following lists items which are considered to fall within the scope of the presentation. Most of the conventions listed follow the IEEE stylistic guidelines for authors (specifically the "IEEE Style Manual" and the "IEEE Information for Authors" guidelines available from the IEEE Author Information Page). Please adhere strictly to the language, formatting and layout specifications for the reports. Referencing is particularly important.

1. **Consistency**: For aspects not specifically mentioned in the rest of this list the guiding principle should be readability, which is greatly influenced by consistency. If a particular style is chosen for an aspect, this style must be applied consistently throughout.

2. **Language**: The report must be compiled in English using formal / technical language.

3. **Spelling**: All relevant modern document generation environments support spell checking. Spelling errors are therefore entirely unacceptable. Use British English, not American.

4. **Sentences**: Use complete and short sentences. Try to keep sentences below a line and a half long.

5. **Layout**: Use single column format for document pages. Ensure that paragraph (full) justification is applied. Do not indent the first line of a paragraph. Avoid half-page and single sentence paragraphs. Margins should be: left 30*mm*, right 25*mm*, top 20*mm* and bottom 20*mm*. When using LaTeX please use the "`geometry`" package to reduce margin whitespace from the default.

6. **Font**: Use 11 point font, 1.5× line spacing and 3× paragraph spacing. The text body font should be Times New Roman or equivalent.

7. **Sections**: Do not underline section titles. Headings should be left justified. Avoid excessive whitespace, do not start each section on a new page. Note that the section titles / headings should be meaningful and generic / uninformative titles such as "Theory" and "Implementation" should be avoided.

8. **Page Numbering**: Include the page number in lower right corner.

9. **Equations**: Equations should be inserted into the text as they are introduced and numbered consecutively. As the equations form part of the text they should be punctuated appropriately. The equation itself should be indented approximately 1 *cm* relative to the left margin (in LaTeX use the "`fleqn`" option to the "`amsmath` package"). The number of the equation must be included within parentheses and aligned on the right margin of the page. Note that not all equations should be numbered. Specifically, the interim steps in a derivation should not normally be number unless the particular step will be referenced in the text. Conversely, if an equation line is numbered it must be referenced somewhere in the text adjacent to the equation. Normal conventions should be followed:
   - letters or symbols representing scalars should be typeset in italics
   - standard mathematical functions (*e.g.*, sin, log) and numeric constants are not italicised
   - matrices should be typeset in upright bold uppercase letters
   - vectors (a column of scalars) in upright bold lowercase letters

   In most subject fields specific letters and symbols are by convention used in published literature to represent specific entities and the same convention should be used as far as possible. As part of the learning process, and to show that you are aware of the conventions, take care to define all such standard usage in your report before the point of first usage.

10. **Figures**: Should always have a caption below the image. Figures should be horizontally centred, whilst the caption should be left aligned relative to the figures. Captions should be punctuated correctly. If the figures are derived from a reference, the caption should include the appropriate citation. **Warning**: although this is acceptable for the unpublished assignment reports, for a published document (*e.g.*, your MEng dissertation and conference / journal papers) you may not include figures from a reference unless you have obtained permission. If at all possible, never place a figure before the first reference to it in the text (for LaTeX, use the `[htbp]` option to the figure environment to set placement priorities). All figures must be referenced and citations to figures in text always carry the abbreviation "Fig." (IEEE convention) followed by the figure number. The abbreviation is used even when it begins a sentence. Figures should fit on one page. Do not use frames around figures. In the case of a block / flow diagram, make the borders of functional units thicker than the flow lines used to connect the blocks to improve readability. Light shading of the functional units is also recommended. Use vector formats and avoid raster images. Raster formats based on lossy compression (such as JPEG) should be avoided unless the figure is a picture.

11. **Plots**: Most of the comments for figures apply. Plots should always have a title and axis labels.

12. **Tables**: Tables should be horizontally centred and should always have a caption. The column and/or row titles should be in bold. Large tables with detailed results should be placed in an appendix, with a corresponding summary table in the main text. Tables in the body of the report should fit on one page.

13. **Acronyms**: Define acronyms the first time they appear. The definition should be written out as part of the sentence, followed by the acronym in parentheses.

14. **Algorithms / Source Code**: Since source code is to be submitted, do not include the source code into your report. Rather give the algorithmic pseudo code in your report and indicated in which source file the algorithm is implemented. Please use the mathematical style pseudo code (for an example, see the Wikipedia page on the Conjugate Gradient Method, specifically the "The resulting algorithm" section").

15. **References**: A list of references in IEEE Transactions format is compulsory (however strict adherence to the IEEE publication abbreviation conventions is not required). If you use an algorithm, quote, technique or theory someone else has developed, you must include a reference and attribute the item to them. Italicize the titles of books, magazines, journals, *etc.* Enclose in quotation marks the titles of journal papers, conference articles and book chapters. Take care to reproduce the title of the reference exactly. The capitalisation of the original title should be retained. Reference the original / authoritative publication as first published. Web / online references should only be used as a last resort and *Wikipedia should never be referenced*.

16. **Footnotes**: Technical documents should avoid footnotes as they disrupt the flow.

A.8 Implementations

Although the postgraduate modules are not software engineering modules, a certain level of quality and format consistency is required in implementations. If applicable, develop reusable code by creating generic functions and modular code. Ensure readability by choosing and using a source code style. Additional requirements include:

1. **Functional description**: Each functional grouping of code should have a block description which describes in words the function of the code group / block.

2. **Block comments**: Commenting each line is generally not useful, the code itself should be written to be readable. It is however advisable to comment a block of code which performs a specific operation. This comment would typically describe aspects of the algorithm implemented and highlight non-obvious assumptions, limitations, side-effects, *etc.*

   Remember that comments are not for describing what the code technically does (that is what the code is for), comments are for what the code is intended to do. Try and comment the decisions you made when developing the code, specifically why you took the approach you did and why you didn't use other options.

   Don't assume that, even six months from now, you're going to remember why you did things a certain way. And the corollary: Don't assume you're going to be the one modifying the code a year or two from now.

3. **Alignment**: Standard practice is to use spaces instead of tabs for alignment. Be sure to set your editor to use 2 or 4 spaces for indentation.

4. **Page width**: Aim for approximately 80 characters per line, no more than 100 per line.

Be careful not to over design. Based on an estimate of the division of marks, plan the execution of the assignments so that you do not to spend disproportionate amounts of time on an aspect that may only count a small amount of marks.

B Originality and Plagiarism

Plagiarism is a serious offence, particularly in an academic environment. The University of Pretoria therefore requires that all material submitted for evaluation be accompanied by a signed Declaration of Originality.

The first page of the submitted materials must therefore be a title page. The second page must contain the declaration of originality and must be signed by the student. The topic of work should correspond with the report title. **Note:** The signature embedded must be a high resolution scanned version of your hand-written signature and not a PDF digital certificate. For your convenience the standard template is provided on the next page of this document. A LaTeX version (`declaration_of_originality.tex`) and a plain text version (`declaration_of_originality.txt`) are available in the top directory of the class directory.

For more information regarding plagiarism and the University's policy, see the UP Plagiarism Prevention section of the Department of Library Services.

DECLARATION OF ORIGINALITY

UNIVERSITY OF PRETORIA

The University of Pretoria places great emphasis upon integrity and ethical conduct in the preparation of all written work submitted for academic evaluation.

While academic staff teach you about referencing techniques and how to avoid plagiarism, you too have a responsibility in this regard. If you are at any stage uncertain as to what is required, you should speak to your lecturer before any written work is submitted.

You are guilty of plagiarism if you copy something from another author's work (*e.g.*, a book, an article or a website) without acknowledging the source and pass it off as your own. In effect you are stealing something that belongs to someone else. This is not only the case when you copy work word-for-word (verbatim), but also when you submit someone else's work in a slightly altered form (paraphrase) or use a line of argument without acknowledging it. You are not allowed to use work previously produced by another student. You are also not allowed to let anybody copy your work with the intention of passing if off as his/her work.

Students who commit plagiarism will not be given any credit for plagiarised work. The matter may also be referred to the Disciplinary Committee (Students) for a ruling. Plagiarism is regarded as a serious contravention of the University's rules and can lead to expulsion from the University.

The declaration which follows must accompany all written work submitted while you are a student of the University of Pretoria. No written work will be accepted unless the declaration has been completed and attached.

Full names of student: _____

Student number: _____

Topic of work: _____

Declaration

1. I understand what plagiarism is and am aware of the University's policy in this regard.
2. I declare that this assignment report is my own original work. Where other people's work has been used (either from a printed source, Internet or any other source), this has been properly acknowledged and referenced in accordance with departmental requirements.
3. I have not used work previously produced by another student or any other person to hand in as my own.
4. I have not allowed, and will not allow, anyone to copy my work with the intention of passing it off as his or her own work.

SIGNATURE: _____ DATE: _____