200645 - PBDE - Statistical Programming and Databases

Coordinating unit: 200 - FME - School of Mathematics and Statistics
Teaching unit: 707 - ESAII - Department of Automatic Control
Academic year: 2017
Degree: MASTER’S DEGREE IN STATISTICS AND OPERATIONS RESEARCH (Syllabus 2013). (Teaching unit Optional)
ECTS credits: 5  Teaching languages: English

Teaching staff
Coordinator: JOAQUIN GABARRÓ VALLÉS
Others: Segon quadrimestre:
JOAQUIN GABARRÓ VALLÉS - A
ALEXANDRE PERERA LLUNA - A

Prior skills
Non compulsory subject.
The student has already developed several abilities in Statistics and/or Operations Research previously.
A B2 (Cambridge First Certificate, TOEFL PBT >550) level of English is required.

Degree competences to which the subject contributes

Specific:
3. CE-1. Ability to design and manage the collection of information and coding, handling, storing and processing it.

4. CE-4. Ability to use different inference procedures to answer questions, identifying the properties of different estimation methods and their advantages and disadvantages, tailored to a specific situation and a specific context.

5. CE-5. Ability to formulate and solve real problems of decision-making in different application areas being able to choose the statistical method and the optimization algorithm more suitable in every occasion.

Translate to English
6. CE-6. Ability to use appropriate software to perform the necessary calculations in solving a problem.

7. CE-7. Ability to understand statistical and operations research papers of an advanced level. Know the research procedures for both the production of new knowledge and its transmission.

8. CE-8. Ability to discuss the validity, scope and relevance of these solutions and be able to present and defend their conclusions.

Transversal:
2. EFFECTIVE USE OF INFORMATION RESOURCES: Managing the acquisition, structuring, analysis and display of data and information in the chosen area of specialisation and critically assessing the results obtained.

10. FOREIGN LANGUAGE: Achieving a level of spoken and written proficiency in a foreign language, preferably English, that meets the needs of the profession and the labour market.

11. TEAMWORK: Being able to work in an interdisciplinary team, whether as a member or as a leader, with the aim of contributing to projects pragmatically and responsibly and making commitments in view of the resources that are available.
This course presents and discusses tools and techniques to prepare students to data science. Main concepts introduced in class will cover tools and methods for data storage and analysis, including relational DB, noSQL and distributed databases, scientific computing, applied machine learning and deep learning with python. The course consists of two main modules.

MODULE 1:
First modulus will cover a crash course for scientific python for data analysis. This crash course will include four main stages:
- Introduction to python language as a tool. ipython, ipython notebook (jupyter), basic types, mutability and immutability and object oriented programming.
- Short introduction to numerical python and matplotlib for graphical visualization.
- Introduction to scientific kits for data analysis with machine learning. Principal components analysis, clustering and supervised analysis with multivariate data.
- Introduction to Deep Learning with Python.

MODULE 2:
The second module develops aspects of data storage. In a more specific way:
- Query an existing relational DB (like PostreSQL).
- Update a current DB and create.
- Data Base management and concurrency problems onDB
- Work with distributed BD like SPARK.

Learning objectives of the subject

This course presents and discusses tools and techniques to prepare students to data science. Main concepts introduced in class will cover tools and methods for data storage and analysis, including relational DB, noSQL and distributed databases, scientific computing, applied machine learning and deep learning with python. The course consists of two main modules.

MODULE 1:
First modulus will cover a crash course for scientific python for data analysis. This crash course will include four main stages:
- Introduction to python language as a tool. ipython, ipython notebook (jupyter), basic types, mutability and immutability and object oriented programming.
- Short introduction to numerical python and matplotlib for graphical visualization.
- Introduction to scientific kits for data analysis with machine learning. Principal components analysis, clustering and supervised analysis with multivariate data.
- Introduction to Deep Learning with Python.

MODULE 2:
The second module develops aspects of data storage. In a more specific way:
- Query an existing relational DB (like PostreSQL).
- Update a current DB and create.
- Data Base management and concurrency problems onDB
- Work with distributed BD like SPARK.

Study load

<table>
<thead>
<tr>
<th>Total learning time: 125h</th>
<th>Hours large group: 30h</th>
<th>24.00%</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Hours small group: 15h</td>
<td>12.00%</td>
</tr>
<tr>
<td></td>
<td>Self study: 80h</td>
<td>64.00%</td>
</tr>
</tbody>
</table>
### Introduction to Python

**Description:**
- Why Python?
- Python History
- Installing Python
- Python resources

**Learning time:** 1h  
Theory classes: 1h

### Working with Python

**Description:**
- Workflow
- ipython vs. CLI
- Text Editors
- IDEs
- Notebook

**Learning time:** 1h  
Theory classes: 1h

### Getting started with Python

**Description:**
- Introduction
- Getting Help
- Basic types
- Mutable and in-mutable
- Assignment operator
- Controlling execution flow
- Exception handling

**Learning time:** 1h  
Theory classes: 1h
| **Functions and Object Oriented Programming** | **Learning time:** 1h  
Theory classes: 1h |
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Description:</strong></td>
<td></td>
</tr>
</tbody>
</table>
- a. Defining Functions  
- b. Input and Output  
- c. Standard Library  
- d. Object-oriented programming |

| **Introduction to NumPy** | **Learning time:** 2h  
Theory classes: 2h |
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Description:</strong></td>
<td></td>
</tr>
</tbody>
</table>
- a. Overview  
- b. Arrays  
- c. Operations on arrays  
- d. Advanced arrays (ndarrays)  
- e. Notes on Performance (%timeit in ipython) |

| **Matplotlib** | **Learning time:** 2h  
Theory classes: 2h |
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Description:</strong></td>
<td></td>
</tr>
</tbody>
</table>
- a. Introduction  
- b. Figures and Subplots  
- c. Axes and Further Control of Figures  
- d. Other Plot Types  
- e. Animations |

| **Python scikits** | **Learning time:** 1h  
Theory classes: 1h |
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Description:</strong></td>
<td></td>
</tr>
</tbody>
</table>
- a. Introduction  
- b. scikit-timeseries |
### scikit-learn

**Description:**
- a. Datasets
- b. Sample generators
- c. Unsupervised Learning
- d. Supervised Learning
  - i. Linear and Quadratic Discriminant Analysis
  - ii. Nearest Neighbors
  - iii. Support Vector Machines
- e. Feature Selection

**Learning time:** 8h
- Theory classes: 8h

### Practical Introduction to Scikit-learn

**Description:**
- a. Solving an eigenfaces problem
  - i. Goals
  - ii. Data description
  - iii. Initial Classes
  - iv. Importing data
- b. Unsupervised analysis
  - i. Descriptive Statistics
  - ii. Principal Component Analysis
  - iii. Clustering
- c. Supervised Analysis
  - i. k-Nearest Neighbors
  - ii. Support Vector Classification
  - iii. Cross validation

**Learning time:** 5h 30m
- Theory classes: 5h 30m

### Introduction to the relational data bases

**Description:**
- Basic concepts on DB like tables, tuples. First steps in PostgreSQL

**Learning time:** 5h
- Theory classes: 5h
<table>
<thead>
<tr>
<th>Topic</th>
<th>Learning time:</th>
<th>Description:</th>
</tr>
</thead>
<tbody>
<tr>
<td>SQL and relational algebra</td>
<td>5h</td>
<td>Queries, insertions and deletions, joins, Elements of the relational algebra. Ordering, grouping, averages.</td>
</tr>
<tr>
<td>Distributed databases</td>
<td>7h 30m</td>
<td>Introduction to Distributed DB SPARK</td>
</tr>
<tr>
<td>Concurrency problems</td>
<td>5h</td>
<td>Concurrent access, Integrity constraints violations, Transactions</td>
</tr>
</tbody>
</table>

**Qualification system**

Final grade will be composed by:
- 1/4 Written exam first module
- 1/4 Written exam first module
- 1/2 Final practical assignment on large databases integrating concepts from both modules
Bibliography

**Basic:**


**Complementary:**
