SECTION DE SYSTEMES DE COMMUNICATION

DE L'ECOLE POLYTECHNIQUE FEDERALE DE LAUSANNE

Master en Data Science

LIVRET DES COURS

ANNEE ACADEMIQUE 2018/2019

TABLE DES MATIERES Page

Le livret des cours est aussi disponible depuis l'adresse internet de la section :

http://ic.epfl.ch/science-donnees

Ordonnance
sur la formation menant au bachelor et au master de l'Ecole polytechnique fédérale de Lausanne
(Ordonnance sur la formation à l'EPFL)

Modification du 30 juin 2015

La Direction de l'Ecole polytechnique fédérale de Lausanne (EPFL)
arrête:

I

L'ordonnance du 14 juin 2004 sur la formation à l'EPFL[1] est modifiée comme suit:

Art. 3, al. 3

Abrogé

Art. 4, al. 2 et 3

2 Les crédits ECTS sont acquis de façon cumulative selon les conditions définies par l'ordonnance du 30 juin 2015 sur le contrôle des études à l'EPFL[2]. Les règlements d'application du contrôle des études visés à l'art. 5 de ladite ordonnance définissent le nombre de crédits attribué à chaque domaine d'études.

3 Les plans d'études visés à l'art. 5 de l'ordonnance sur le contrôle des études à l'EPFL sont conçus de façon à permettre l'acquisition de 60 crédits ECTS par année académique.

Art. 5 Nombre de crédits ECTS requis

1 A réussi le bachelor l'étudiant qui a acquis 180 crédits ECTS conformément à l'ordonnance du 30 juin 2015 sur le contrôle des études à l'EPFL[3] et aux règlements d'application visés à l'art. 5 de ladite ordonnance.

2 A réussi le master l'étudiant qui a acquis, en sus du bachelor, 60 crédits ECTS, respectivement 90 crédits ECTS pour les sections qui les requièrent conformément à l'annexe I, et réussi le projet de master représentant 30 crédits, conformément à l'ordonnance sur le contrôle des études à l'EPFL et aux règlements d'application.

Art. 6, al. 2

Abrogé

1 RS **414.132.3**
2 RS **414.132.2**
3 RS **414.132.2**

Art. 7, al. 1

[1] Le cycle propédeutique s’étend sur deux semestres.

Art. 8, al. 3 et 4

[3] Il doit être réussi au plus tard quatre ans après la réussite du cycle propédeutique ou, en cas d’admission à un semestre supérieur, dans un délai qui correspond au double du nombre de semestres à accomplir.

[4] Le cycle bachelor est réputé réussi par l’acquisition de 120 crédits ECTS. La réussite du cycle bachelor est la condition pour entrer au cycle master. L’art. 29, al. 1, de l’ordonnance du 30 juin 2015 sur le contrôle des études à l’EPFL[4] est réservé.

Art. 9, al. 2

Abrogé

Art. 11 Projet de master

[1] Le projet de master s’étend sur un semestre et sa réussite permet d’acquérir 30 crédits ECTS.

[2] Le projet de master doit être réussi dans le délai d’un an après la réussite du cycle master ou, le cas échéant, après l’admission conditionnelle (art. 29, al. 3, de l’ordonnance du 30 juin 2015 sur le contrôle des études à l’EPFL[5]).

[3] La réussite du cycle master est la condition pour entamer le projet de master. L’art. 29, al. 3, de l’ordonnance du 30 juin 2015 sur le contrôle des études à l’EPFL est réservé; s’il s’applique, la réussite du projet de master implique la réussite préalable du cycle master.

Art. 12 Conditions liées aux durées

[1] Les crédits requis doivent être acquis dans les durées fixées pour chaque cycle de formation par la présente ordonnance.

[2] En dérogation à l’al. 1, l’école peut prolonger la durée maximale d’un cycle de formation ou accorder une interruption entre deux cycles à un étudiant qui fait valoir un motif valable, notamment une longue maladie, une maternité, une période d’obligation de servir, dès qu’il en a connaissance et avant l’échéance de la durée maximale.

Art. 13, al. 2

[2] Les directives de l’école s’appliquent.

[4] RS **414.132.2**
[5] RS **414.132.2**

II

La présente ordonnance entre en vigueur le 1er septembre 2016.

30 juin 2015

Au nom de la direction
de l'Ecole polytechnique fédérale de Lausanne:

Le président, Patrick Aebischer
Le General Counsel, Susan Killias

Ordonnance sur le contrôle des études menant au bachelor et au master à l'Ecole polytechnique fédérale de Lausanne
(Ordonnance sur le contrôle des études à l'EPFL)

du 30 juin 2015

La Direction de l'Ecole polytechnique fédérale de Lausanne (Direction de l'EPFL),
vu l'art. 3, al. 1, let. b, de l'ordonnance du 13 novembre 2003 sur l'EPFZ et l'EPFL[1],
arrête:

Chapitre 1 Dispositions générales
Section 1 Objet et champ d'application

Art. 1 Objet

La présente ordonnance arrête les règles de base du contrôle des études à l'EPFL.

Art. 2 Champ d'application

1 La présente ordonnance s'applique à la formation menant au bachelor et au master de l'EPFL.

2 Dans la mesure où la direction de l'EPFL n'a pas édicté de règles particulières, les art. 8, 10, 12, 14, 15 et 18 à 20 s'appliquent également:

a. aux examens d'admission;
b. aux examens du cours de mathématiques spéciales (CMS);
c. aux examens du cours de mise à niveau;
d. aux examens de doctorat;
e. aux examens des programmes doctoraux;
f. aux examens de la formation continue et de la formation approfondie.

Section 2 Définitions générales

Art. 3 Branche

1 Une branche est une matière d'enseignement faisant l'objet d'une ou de plusieurs épreuves.

RS **414.132.2**
1 RS **414.110.37**

[2] Une branche dite de semestre est une branche dont les épreuves se déroulent pendant la période de cours.

[3] Une branche dite de session est une branche dont une épreuve se déroule en session d'examens. Elle peut comporter des épreuves se déroulant pendant la période de cours.

[4] Une branche de semestre peut consister en un stage.

Art. 4 Crédits et coefficients

A toute branche est associé un nombre de crédits ECTS (European Credit Transfer and Accumulation System) (crédits) ou, pour le cycle propédeutique, un coefficient, qui indiquent son poids dans la formation.

Section 3
Dispositions communes aux études de bachelor et de master

Art. 5 Plans d'études et règlements d'application

Des plans d'études et des règlements d'application sont édictés pour chaque cycle d'études de chaque domaine. Ils définissent en particulier:

a. les branches de semestre et les branches de session;
b. le semestre ou la session pendant lesquels ces branches peuvent être présentées;
c. la forme (écrite ou orale) de l'épreuve en session;
d. la composition des blocs et des groupes de branches;
e. les coefficients ou les crédits attribués à chaque branche;
f. le nombre de crédits ou le coefficient à acquérir dans chaque bloc et chaque groupe;
g. les conditions applicables aux prérequis (art. 25);
h. les conditions de réussite particulières;
i. les études d'approfondissement, de spécialisation ou interdisciplinaires;
j. les éventuels régimes transitoires applicables aux modifications des plans d'études et des règlements d'application.

Art. 6 Blocs et groupes de branches

[1] Les branches sont rassemblées en bloc ou en groupe. Chaque branche ne peut faire partie que d'un seul bloc ou d'un seul groupe. Un bloc peut être constitué d'une seule branche.

[2] Un bloc est réputé réussi:

a. lorsque la somme des crédits acquis par branche est égale ou supérieure au nombre requis; ou
b. lorsque la somme des crédits acquis pour les branches présentées atteint le nombre requis et que la moyenne du bloc (art. 8, al. 5) est égale ou supérieure à 4,00; dans ce cas, la totalité des crédits des branches présentées est acquise.

[3] Un groupe est réputé réussi lorsque les crédits des branches qui le composent ont été accumulés jusqu'au nombre requis; aucune compensation n'est possible entre les notes des branches du groupe.

Art. 7 Fiches de cours

Les fiches de cours publiées indiquent en particulier, pour chaque branche:

a. les objectifs de formation;
b. un bref descriptif de la matière;
c. les épreuves composant la note finale, avec leur pondération et leur forme;
d. les éventuels prérequis (art. 25);
e. la langue d'enseignement.

Art. 8 Notation

[1] Une épreuve est notée de 1,00 à 6,00. Les notes en dessous de 4,00 sanctionnent des prestations insuffisantes. L'épreuve est notée 0 lorsque l'étudiant ne se présente pas, ne répond à aucune question ou ne respecte pas les délais.

[2] La note finale de la branche se compose des notes de ses épreuves. Elle est arrêtée au quart de point. Lorsqu'elle est inférieure à 1,00, la branche est considérée comme non acquise et notée NA (non acquis). L'appréciation NA compte comme tentative de réussite.

[3] Le règlement d'application peut prévoir qu'une branche est notée au moyen des appréciations R (réussi) ou E (échec).

[4] Lorsque la branche est répétée, la note retenue est celle de la seconde tentative.

[5] Les moyennes sont calculées en pondérant chaque note finale chiffrée de branche par son coefficient ou son nombre de crédits. Elles sont arrêtées au centième. Les appréciations NA et E empêchent l'obtention d'une moyenne, sauf dans les cas visés à l'art. 6, al. 2, let. b, et 3.

Art. 9 Organisation des sessions et des épreuves et inscriptions aux branches

[1] Deux sessions d'examens sont organisées par année académique. Elles ont lieu entre les semestres.

2 Les délais d'inscription aux branches, les délais de retrait, les horaires et les dates des épreuves, ainsi que les autres modalités sont communiqués aux étudiants.

3 A l'échéance des délais, les inscriptions aux branches et les retraits sont définitifs.

4 Lorsque l'étudiant répète une branche, celle-ci est régie par les dispositions en vigueur au moment de la répétition, à moins que l'école n'en ait disposé autrement.

Art. 10 Incapacité

1 L'étudiant qui se prévaut d'un motif d'incapacité à se présenter à une épreuve doit l'annoncer à l'école dès la survenance de ce motif.

2 Il lui présente en outre les pièces justificatives au plus tard trois jours après la survenance du motif d'incapacité. Par pièces justificatives, on entend notamment un certificat médical ou une attestation d'une obligation légale de servir.

3 Invoquer un motif d'incapacité après s'être présenté à l'épreuve ne justifie pas l'annulation d'une note.

Art. 11 Langue des épreuves

1 Les épreuves se déroulent dans la langue de l'enseignement de la branche.

2 L'étudiant a le droit de répondre en français à une épreuve en anglais. Sur demande écrite de sa part, l'enseignant peut lui accorder de répondre en anglais si l'épreuve est en français.

Art. 12 Etudiants en situation de handicap

1 Si un candidat en situation de handicap en fait la demande au début de l'année académique, l'école fixe un déroulement d'épreuve adapté à son handicap et décide de l'utilisation de moyens auxiliaires ou de l'assistance personnelle nécessaires.

2 Les objectifs de l'épreuve doivent être garantis.

Art. 13 Tâches de l'enseignant

1 L'enseignant remplit notamment les tâches suivantes:

a. donner les informations nécessaires sur ses matières d'enseignement pour qu'elles soient publiées dans la fiche de cours;

b. informer les étudiants, s'il y a lieu, du contenu des matières et du déroulement des épreuves;

c. conduire les épreuves;

d. prendre des notes de chaque épreuve orale, qu'il peut être appelé à produire auprès de la conférence d'examen ou des autorités de recours;

e. attribuer les notes des épreuves, ainsi que la note finale de branche;

f. conserver pendant six mois après la fin du cycle concerné (chap. 2 à 4) les épreuves écrites et les notes prises durant les épreuves orales; en cas de recours, ce délai est prolongé jusqu'au terme de la procédure.

2 S'il est empêché de remplir ses tâches, le directeur de section désigne un remplaçant.

Art. 14 Observateur

1 Un observateur désigné par le directeur de section assiste à l'épreuve orale ayant lieu en session d'examens, dans le but de veiller à son déroulement régulier.

2 Il prend, pour chaque candidat, des notes sur le déroulement de l'épreuve et les conserve conformément à l'art. 13, al. 1, let. f.

Art. 15 Consultation des épreuves

L'étudiant peut consulter son épreuve dans les 6 mois qui suivent la communication du résultat.

Art. 16 Commissions d'évaluation

Des commissions d'évaluation peuvent être mises sur pied pour les branches de semestre. Outre l'enseignant et un expert, les commissions d'évaluation peuvent comprendre les assistants et les chargés de cours qui ont participé à l'enseignement, ainsi que d'autres professeurs.

Art. 17 Conférence d'examen

1 La conférence d'examen siège à l'issue de chaque session. Elle est composée du vice-provost pour la formation, qui la préside, du directeur de section et du chef du service académique. Les membres de la conférence d'examen peuvent se faire représenter par leur suppléant.

2 La conférence d'examen se prononce sur les cas particuliers conformément aux dispositions légales.

Art. 18 Fraude

1 Par fraude, on entend toute forme de tricherie en vue d'obtenir pour soi-même ou pour autrui une évaluation non méritée.

2 En cas de fraude, de participation à la fraude ou de tentative de fraude, le règlement disciplinaire du 15 décembre 2008 concernant les étudiants de l'Ecole polytechnique fédérale de Lausanne[2] s'applique.

2 RS **414.138.2**

Art. 19 Notification des résultats et communications

[1] La décision de réussite ou d'échec pour le cycle d'études est notifiée à l'étudiant.

[2] Elle fait mention des notes obtenues et des crédits acquis.

[3] La notification de la décision ainsi que les communications ont lieu par voie électronique ou postale.

Art. 20 Demande de nouvelle appréciation et recours administratif

[1] La décision peut faire l'objet d'une demande de nouvelle appréciation auprès de l'école dans les 10 jours qui suivent sa notification. L'art. 63, al. 1, 3 et 4, de la loi fédérale du 20 décembre 1968 sur la procédure administrative[3] est applicable.

[2] Elle peut également faire l'objet d'un recours administratif auprès de la commission de recours interne des EPF, dans les 30 jours qui suivent sa notification.

Chapitre 2 Examens du cycle propédeutique

Art. 21 Conditions de réussite

[1] L'étudiant qui, à l'issue du premier semestre du cycle propédeutique et de la session d'examens afférente, a atteint une moyenne pondérée (art. 8, al. 5) d'au moins 3,50 pour le premier bloc au sens du règlement d'application est admis au second semestre du cycle.

[2] A réussi le cycle propédeutique l'étudiant qui, conformément au plan d'études et au règlement d'application:

a. a présenté toutes les branches; et

b. a obtenu une moyenne égale ou supérieure à 4,00 dans chacun des blocs et, le cas échéant, les coefficients requis dans un groupe.

Art. 22 Echec et élimination

[1] Constituent un échec, au niveau du cycle propédeutique:

a. la non-atteinte d'une moyenne pondérée d'au moins 3,50 pour le premier bloc, à l'issue du premier semestre et de la session d'examens afférente;

b. la non-atteinte d'une moyenne pondérée d'au moins 4,00 par bloc ou la non-atteinte du nombre de coefficients requis dans un groupe, à l'issue du cycle propédeutique; ou

c. le fait de ne pas avoir présenté toutes les branches du cycle propédeutique, sous réserve de l'art. 23, al. 4.

3 RS **172.021**

2 L'étudiant qui suit le cycle propédeutique en première tentative et se trouve dans la situation visée à l'al. 1, let. a, suit au second semestre le cours de mise à niveau de l'EPFL.

3 Est assimilé à un échec au cycle propédeutique de l'EPFL un échec ou une absence de réussite subi dans une autre haute école à un niveau comparable au cycle propédeutique, si la majorité des branches sont considérées par l'EPFL comme étant analogues.

4 Constitue un échec définitif un second échec au niveau du cycle propédeutique ou le non-respect de la durée maximale de deux ans pour réussir le cycle.

5 Constituent un motif d'exclusion définitive de toute formation de bachelor à l'EPFL la non-atteinte d'une moyenne pondérée d'au moins 4,00 à l'issue du cours de mise à niveau ou le non-respect de l'obligation de le suivre.

Art. 23 Répétition

1 L'étudiant qui est en situation d'échec, en première tentative, selon l'art. 22, al. 1, let. b et c, ou qui a atteint une moyenne d'au moins 4,00 au cours de mise à niveau est admis une seconde fois au premier semestre du cycle propédeutique de l'année académique qui suit.

2 Les branches d'un bloc ou d'un groupe réussis (art. 21, al. 2, let. b) sont acquises et ne peuvent pas être répétées.

3 La répétition des autres branches non réussies est impérative. La répétition des branches réussies est facultative, sauf pour les étudiants issus de la situation visée à l'art. 22, al. 1, let. a, pour lesquels elle est obligatoire. Le règlement d'application peut toutefois prévoir que certaines branches de semestre réussies ne peuvent pas être répétées.

4 En cas d'absence justifiée au sens de l'art. 10, l'école examine s'il est raisonnablement exigible de l'étudiant qu'il complète le cycle propédeutique à la session ordinaire correspondante de l'année suivante ou si l'étudiant doit être considéré comme ayant échoué.

Chapitre 3 Examens du cycle bachelor et du cycle master

Art. 24 Crédits

Les crédits de la branche sont attribués lorsque la note obtenue est égale ou supérieure à 4,00 ou que la moyenne du bloc de branches à laquelle elle appartient est égale ou supérieure à 4,00.

Art. 25 Prérequis

Le règlement d'application ou la fiche de cours définit les branches dont l'étudiant doit avoir acquis les crédits afin d'être admis à suivre d'autres branches.

Art. 26 Conditions de réussite

[1] Les crédits requis du cycle bachelor et du cycle master doivent être acquis conformément à la présente ordonnance, à l'ordonnance du 14 juin 2004 sur la formation à l'EPFL[4] et au règlement d'application.

[2] Dans le cycle bachelor, 60 crédits au moins doivent être acquis par tranche de deux ans.

Art. 27 Répétition

[1] Si, dans un bloc ou un groupe, le nombre de crédits requis n'est pas acquis, les branches dont la note est inférieure à 4,00 peuvent être répétées une fois, impérativement à la session ordinaire de l'année qui suit.

[2] L'étudiant qui échoue deux fois à une branche optionnelle peut en présenter une nouvelle conformément au plan d'études.

Art. 28 Echec définitif

Si l'étudiant n'acquiert pas les crédits requis conformément à la présente ordonnance et au règlement d'application, dans le respect des durées maximales fixées par l'ordonnance du 14 juin 2004 sur la formation à l'EPFL[5], il se trouve en situation d'échec définitif.

Art. 29 Admission conditionnelle au cycle consécutif

[1] Peut être admis conditionnellement au cycle master consécutif l'étudiant qui:

a. n'a pas plus de 10 crédits manquants sur ceux requis par le plan d'études de dernière année du cycle bachelor de l'EPFL; et

b. n'est pas en situation d'échec définitif.

[2] L'étudiant admis conditionnellement au cycle master consécutif a l'obligation d'acquérir les crédits manquants du bachelor dans l'année de son admission conditionnelle, sous peine d'être exclu du cycle.

[3] Peut être admis conditionnellement au projet de master l'étudiant qui:

a. n'a pas plus de 8 crédits manquants sur ceux requis pour le cycle master y compris les études visées à l'art. 5, let. i;

b. n'est pas en situation d'échec définitif.

4 RS **414.132.3**
5 RS **414.132.3**

Chapitre 4 Projet de master

Art. 30 Déroulement

1 Le sujet du projet de master est fixé ou approuvé par le professeur ou le maître d'enseignement et de recherche qui en assume la direction.

2 Sur demande, le directeur de section peut confier la direction du projet de master à un professeur ou un maître d'enseignement et de recherche rattaché à une autre section ou à un collaborateur scientifique.

3 L'examen du projet de master consiste en une évaluation de sa présentation finale suivie d'une interrogation orale devant l'enseignant qui a dirigé le projet et un expert externe à l'EPFL désigné par l'enseignant en accord avec le directeur de section. Seul l'enseignant peut inviter d'autres personnes à l'interrogation orale; celles-ci ne participent pas à l'évaluation.

4 Si la qualité rédactionnelle du projet est jugée insuffisante, l'enseignant peut exiger que l'étudiant y remédie dans un délai de deux semaines à compter de l'interrogation orale.

Art. 31 Conditions de réussite

1 Le projet de master est réputé réussi lorsque la note attribuée est égale ou supérieure à 4,00.

2 Si le règlement d'application prévoit un stage associé au projet de master, celui-ci doit avoir été réussi préalablement.

Art. 32 Répétition

1 En cas d'échec, un nouveau projet de master peut être présenté dans le respect de la durée maximale prévue par l'ordonnance du 14 juin 2004 sur la formation à l'EPFL[6].

2 Un second échec constitue un échec définitif.

Chapitre 5 Dispositions finales

Art. 33 Abrogation

L'ordonnance du 14 juin 2004 sur le contrôle des études à l'EPFL[7] est abrogée.

6 RS **414.132.3**
7 RO **2004** 4323, **2006** 4125, **2008** 3721

Art. 34 Disposition transitoire

Le chapitre 2 de l'ordonnance du 14 juin 2004 sur le contrôle des études à l'EPFL[8] demeure applicable jusqu'au 31 août 2017 aux étudiants répétant le cycle propédeutique durant l'année académique 2016–2017.

Art. 35 Entrée en vigueur

La présente ordonnance entre en vigueur le 1er septembre 2016.

30 juin 2015

Au nom de la direction
de l'Ecole polytechnique fédérale de Lausanne:

Le président, Patrick Aebischer
Le General Counsel, Susan Killias

8 RO **2004** 4323, **2006** 4125, **2008** 3721

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Plan d'études

Master en Data Science

2018-2019

arrêté par la direction de l'EPFL le 11 juin 2018

Directeur de la section	**Prof. M. Gastpar**
Adjointe de la section	**Mme S. Dal Mas**
Conseiller d'études :	
1ère année cycle Master	**Prof. R. Guerraoui**
2ème année cycle Master	**Prof. B. Rimoldi**
Coordination des stages en industrie	**Mme S. Dal Mas**
Secrétariat de la section	**Mme C. Dauphin**

Aux cycles bachelor et master, selon les besoins pédagogiques, les heures d'exercices mentionnées dans le plan d'études pourront être intégrées dans les heures de cours ; les scolarités indiquées représentent les nombres moyens d'heures de cours et d'exercices hebdomadaires sur le semestre.

2018-2019 **DATA SCIENCE** **Cycle Master**

Code	Matières	Enseignants sous réserve de modification	Sections	Semestres MA1 c	e	p	MA2 c	e	p	Crédits	Période des épreuves	Type examen
	Groupe "Core courses et options"									**72**		
	Groupe 1 "Core courses"									**min. 30**		
CS-450	Advanced Algorithms	Svensson	IN				**4**	**2**	**1**	7	E	écrit
CS-401	Applied Data Analysis	West	IN	**2**	**2**					6	H	écrit
COM-402	Information security and privacy	Hubaux/Oechslin/Troncoso	SC/IN				**4**		**2**	6	E	écrit
COM-406	Information Theory and Signal Processing	Gastpar / Telatar / Urbanke	SC	**4**	**2**					6	H	écrit
CS-433	Machine learning	Jaggi / Urbanke	IN / SC	**4**	**2**					7	H	écrit
CS-439	Optimization for Machine Learning	Jaggi	IN				**2**	**2**		4	E	écrit
MATH-413	Statistics for Data Science	Panaretos	MA	**4**	**2**					6	H	écrit
CS-449	Systems for Data Science	Koch	IN				**2**	**2**	**2**	6	sem P	
	Groupe 2 "Options"	**(la somme des crédits des groupes 1 et 2 doit être de 72 crédits au minimu**										
---	Cours à option	Divers enseignants	Divers									
	Bloc "Projets et SHS" :									**18**		
COM-418	Projet de semestre en data science	divers enseignants	SC	←		**2**			→	12	sem A ou P	
HUM-nnn	SHS : introduction au projet	divers enseignants	SHS	**2**		**1**				3	sem A	
HUM-nnn	SHS : projet	divers enseignants	SHS						**3**	3	sem P	
	Total des crédits du cycle master									**90**		

Stage d'ingénieur :

Voir les modalités dans le règlement d'application

Mineurs :

Le cursus peut être complété par un des mineurs figurant dans l'offre de l'EPFL (renseignements à la page sac.epfl.ch/mineurs),
à l'exclusion des mineurs"Data Science", "Informatique", "Cyber security" et "Systèmes de communication" qui ne peuvent pas être choisi.
Parmi les mineurs offerts par l'EPFL, la section recommande à ses étudiants les mineurs suivants :
- Biocomputing (SIN)
- Computational Science and Engineering (SMA)
- Management de la technologie et entrepreneuriat (SMTE)
- Technologies biomédicales (SMT)
- Technologies spatiales (SEL)

Le choix des cours de tous les mineurs se fait sur conseil de la section de l'étudiant et du responsable du mineur.

2018-2019 DATA SCIENCE - Options Cycle Master

Code	Matières	Enseignants sous réserve de modification	Sections	Semestres MA1 c	MA1 e	MA1 p	MA2 c	MA2 e	MA2 p	Crédits	Période des épreuves	Type examen	Cours biennaux donnés en
EE-558	A Network Tour of Data Science	Vandergheynst/Frossard	EL	2	2					4	sem A		
COM-501	Advanced cryptography	Vaudenay	SC				2	2		4	E	écrit	
COM-417	Advanced probability and applications	Lévêque	SC				3	2		6	E	écrit	
CS-523	Advanced topics on privac enhancing technologies	Hubaux/Troncoso	SC/IN	3		3				7	A	écrit	
CS-435	Analytic algorithms	Vishnoi	IN				2	1		4	sem P		
MATH-493	Applied biostatistics	Goldstein	MA				2	2		5	sem P		
CS-456	Artificial neural networks	Gerstner	IN				2	1		4	E	écrit	
COM-415	Audio and acoustic signal processing	Faller/Kolundzija	SC	2	2	1				4 5	H	écrit	
EE-592	Automatic speech processing	Bourlard	EL	2	1					3	H	écrit	
BIO-465	Biological modeling of neural networks	Gerstner	IN				2	2		4	E	écrit	
MATH-460	Combinatorial optimization	Eisenbrand	MA	2	2					5	H	écrit	
MATH-453	Computational linear algebra	Massei	MA				2	2		5	E	oral	
CS-523	Computational complexity	Svensson	IN	3	1					4	sem A		2018-2019
CS-413	Computational Photography (pas donné en 2018-2019)	Süsstrunk	SC				2		2	5	sem P		
CS-442	Computer vision	Fua	IN				2	1		4	E	écrit	
CS-454	Convex optimization and applications	Lebret	MTE				1	2		4	sem P		
COM-401	Cryptography and security	Vaudenay	SC	4	2					7	H	écrit	
COM-480	Data Visualization	Benzi	SC	2		2				4	sem A		
CS-411	Digital education & learning analytics	Dillenbourg/Jermann	IN	2		2				4	H	oral	
CS-423	Distributed information systems	Aberer	IN				2	1		4	E	écrit	
ENG-466	Distributed intelligent systems	Martinoli	SIE	2	3					5	H	écrit sans retrait	
CS-525	Foundations and tools for processing tree structured data	Vanoirbeek	IN	2		2				4	A	écrit	
MATH-360	Graph Theory	vacat	MA				2	2		5	E	écrit	
CS-486	Human-computer interaction	Pu	IN				2	1	1	4	sem P		
EE-451	Image analysis and pattern recognition	Thiran J.-P.	EL				2		2	4	sem P		
COM-404	Information theory and coding	Telatar	SC	4	2					7	H	écrit	
CS-430	Intelligent agents	Faltings	IN	3	3					6	sem A		
CS-431	Introduction to natural language processing	Chappelier/Rajman	IN	2	2					4	H	écrit	
EE-490h	Lab in Data Science	Verscheure	SC						4	4	sem P	sans retrait	
CS-526	Learning theory	Macris/Urbanke/Svensson	SC/IN				2	2		4	E	écrit	
MATH-341	Linear models	Thibaud	MA	2	2					5	H	écrit	
COM-516	Markov chains and algorithmic applications	Lévêque/Macris	SC	2	2					4	A	écrit	
COM-514	Mathematical foundations of signal processing	Kolundzija/Scholefield/Parhizkar	SC	3	2					6	H	écrit	
EE-556	Mathematics of data: from theory to computation	Cevher	EL	2	2					4	sem A		
COM-512	Networks out of control (pas donné en 2018-2019)	Thiran P./Celis	SC				2	1		4	E	écrit	2019-2020
COM-508	Optional project in Data Science	Divers enseignants	SC			2				8	sem A ou P		
COM-503	Performance evaluation	Le Boudec	SC				3	1	2	7	E	écrit	2018-2019
MATH-447	Risk, rare events and extremes	Davison	MA	2	2					5	H	écrit	2018-2019
MATH-441	Robust and nonparametric Statistics	Morgenthaler	MA	2	2					5	H	oral	2017-2018
MATH-442	Statistical Theory	Dehaene	MA	2	2					5	E	écrit	
COM-506	Student seminar : Security protocols and applications	Oechslin/Vaudenay	SC				2			3	E	écrit	
CS-410	Technology Ventures in IC (pas donné en 2018-2019)	Bugnion	IN				2		2	4	sem P		
MATH-342	Time Series	Thiebaud	MA				2	2		5	E	écrit	
CS-455	Topics in theoretical computer science (pas donné en 2018-2019)	Svensson	IN	3	1					4	sem A		2019-2020
CS-444	Virtual reality	Boulic	IN				2	1		4	sem P		

2018-2019 **Data Science** **Section de Systèmes de communication**
Mineur disciplinaire **Responsable : Sylviane Dal Mas**

Les enseignants, les crédits et la période des cours sont indiqués sous réserve de modification. **107** **crédits offerts**
Les cours déjà suivis au bachelor ou au master ne peuvent pas être pris également dans un mineur.

Codes	Matières (liste indicative)	Enseignants	Livret des cours	Crédits	Période des cours	
EE-558	A network tour of Data science	Vandergheynst/Frossard	EL	4	A	
CS-450	Advanced algorithms	Svensson	IN	7		P
COM-501	Advanced cryptography	Vaudenay	SC	4		P
COM-417	Advanced probability and applications	Lévêque	SC	6		P
MATH-474	Applied biostatistics	Goldstein	MA	5		P
CS-401	Applied data analysis	West	IN	6	A	
COM-415	Audio and acoustic signal processing	Faller/Kolundzija	SC	5	A	
MATH-435	Bayesian computation	Dehaene	MA	5		P
CS-442	Computer vision	Fua	IN	4		P
COM-480	Data visualization	Benzi	SC	4	A	
COM-402	Information security and privacy	Hubaux/Oechslin/Troncoso	SC/IN	6		P
COM-406	Information theory and signal processing	Gastpar/Telatar/Urbanke	SC	6	A	
CS-430	Intelligent agents	Faltings	IN	6	A	
CS-433	Machine learning	Jaggi/Urbanke	IN/SC	7	A	
COM-512	Network out of control pas donné en 2018-2019	Celis/Thiran	SC	4		P
CS-439	Optimization for machine learning	Jaggi	IN	4		P
COM-503	Performance evaluation	Le Boudec	SC	7		P
MATH-447	Risk, rare events and extremes	Davison	MA	5	A	
MATH-413	Statistics for Data Science	Panaretos	MA	6	A	
CS-449	Systems for Data Science	Koch	IN	6		P

Légende :
A = automne, P = printemps
1 semestre comprend 14 semaines.

RÈGLEMENT D'APPLICATION DU CONTRÔLE DES ÉTUDES DE LA SECTION DE SYSTÈMES DE COMMUNICATION POUR LE MASTER EN DATA SCIENCE pour l'année académique 2018-2018
du 11 juin 2018

La direction de l'École polytechnique fédérale de Lausanne

vu l'ordonnance sur la formation menant au bachelor et au master de l'EPFL du 14 juin 2004,
vu l'ordonnance sur le contrôle des études menant au bachelor et au master à l'EPFL du 30 juin 2015,
vu le plan d'études de la section de systèmes de communication pour le master en Data Science.

arrête:

Article premier - Champ d'application

Le présent règlement fixe les règles d'application du contrôle des études de master de la section de systèmes de communication pour le master en Data Science qui se rapportent à l'année académique 2018-2019.

Art. 2 – Étapes de formation

Le master en Data Science est composé de deux étapes successives de formation :
- le cycle master d'une durée de 3 semestres dont la réussite implique l'acquisition de 90 crédits, condition pour effectuer le projet de master.
- le projet de master, d'une durée de 17 semaines à l'EPFL ou de 25 semaines hors EPFL (industrie ou autre haute école) et dont la réussite se traduit par l'acquisition de 30 crédits. Il est placé sous la responsabilité d'un professeur ou MER affilié à la section de systèmes de communication ou d'informatique.

Art 3 – Sessions d'examen

1. Les branches d'examen sont examinées par écrit ou par oral pendant les sessions d'hiver ou d'été. Elles sont mentionnées dans le plan d'études avec la mention H ou E.

2. Les branches de semestre sont examinées pendant le semestre d'automne ou le semestre de printemps. Elles sont mentionnées dans le plan d'études avec la mention sem A ou sem P.

3. Une branche annuelle, c'est à dire dont l'intitulé tient sur une seule ligne dans le plan d'étude, est examinée globalement pendant la session d'été (E).

4 Pour les branches de session, la forme écrite ou orale de l'examen indiquée pour la session peut être complétée par des contrôles de connaissances écrits ou oraux durant le semestre, selon indications de l'enseignant.

Art. 3 – Prérequis

Certains enseignements peuvent exiger des prérequis qui sont mentionnés dans la fiche de cours concerné. Le cours prérequis est validé si les crédits correspondants ont été acquis pour le cours ou par moyenne du bloc.

Art. 4 – Conditions d'admission

1. Les étudiants issus du Bachelor en Informatique ou en Systèmes de communications sont admis automatiquement.

2. Les étudiants issus du Bachelor en Informatique ou en Systèmes de communication qui n'auront pas fait les cours prérequis durant leur cycle Bachelor devront les faire en parallèle à leur cycle Master.

3. Pour les autres étudiants, l'admission s'effectue sur dossier.

Art. 5 - Organisation

1. Les enseignements du cycle master sont répartis en deux groupes et un bloc dont les crédits doivent être obtenus de façon indépendante.

2. Le bloc « Projets et SHS » est composé d'un projet de 12 crédits et de l'enseignement SHS.

3. Le groupe 1 « Core courses » est composé des cours de la liste du plan d'études dans la rubrique « Master ».

4. Le groupe 2 « Options » est composé
   - des cours de la liste du groupe 2 « options » du plan d'études dans la rubrique « Master » ;
   - des crédits surnuméraires obtenus dans le groupe 1 « Core courses » ;
   - d'un projet optionnel de 8 crédits ;
   - de cours hors plan d'études suivant l'alinéa 6.

5. Le projet du bloc « Projets et SHS » et le projet optionnel du groupe 2 ne peuvent être effectués dans le même semestre.

6. Des cours, comptant pour un maximum de 15 crédits au total, peuvent être choisis en dehors de la liste des cours du plan d'études dans la rubrique « Master ». Le choix de ces cours doit être accepté préalablement par le directeur de la section qui peut augmenter le maximum de 15 crédits si la demande est justifiée.

Art. 6 - Examen du cycle master

1. Le bloc « Projets et SHS » est réussi lorsque **18 crédits** sont obtenus.

2. Le groupe « Core courses et Options », composé du groupe 1 « Core courses » et du groupe 2 « Options » est réussi lorsque **72 crédits** sont obtenus.

3. Le groupe 1 « Core courses » est réussi lorsqu'**au moins 30 crédits** sont obtenus.

Art. 7 - Enseignement SHS

Les deux branches SHS donnent chacune lieu à 3 crédits. L'enseignement du semestre d'automne introduit à la réalisation du projet du semestre de printemps. Pour autant qu'il considère que le motif est justifié, le Collège des Humanités peut déroger à cette organisation. Il peut également autoriser à ce qu'un étudiant réalise son projet sur un semestre qui ne suit pas immédiatement celui dans lequel a lieu l'enseignement d'introduction.

Art. 8 – Mineurs

1. Afin d'approfondir un aspect particulier de sa formation ou de développer des interfaces avec d'autres sections, l'étudiant peut choisir la formation offerte dans le cadre d'un mineur figurant dans l'offre de l'EPFL.

2. Le choix des cours qui composent un mineur se fait avec la section de systèmes de communication et avec le responsable du mineur. Les mineurs, « Data Science » « Informatique », « Cyber Security » et « Systèmes de Communication » ne peuvent pas être choisis.

3. L'étudiant annonce le choix d'un mineur à sa section au plus tard à la fin du premier semestre des études de master.

4. Un mineur est réussi quand 30 crédits au minimum sont obtenus parmi les branches avalisées.

Art. 8 – Stage d'ingénieur

1. Les étudiants commençant leur cycle master doivent effectuer un stage d'ingénieur durant leur master :
- soit un stage d'été de minimum 8 semaines
- soit un stage de minimum 6 mois en entreprise (en congé durant un semestre)
- soit un Projet de Master de 25 semaines en entreprise (valide le stage et le Projet de Master)

2. Le stage peut être effectué dès le 2ème semestre du cycle master, mais avant le projet de master.

3. Le responsable du stage de la section évalue le stage, par l'appréciation « réussi » ou « non réussi ». Sa réussite est une condition pour l'admission au projet de master. En cas de non réussite, il peut être répété une fois, en règle générale dans une autre entreprise.

4. Il est validé avec les 30 crédits du projet de master.

5. Les modalités d'organisation et les critères de validation du stage font l'objet d'une directive interne à la section.

Au nom de la direction de l'EPFL

Le président, M. Vetterli
Le vice-président pour l'éducation, P. Vandergheynst

Lausanne, le 11 juin 2018

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

DATA SCIENCE

Cycle

Master

2018 / 2019

EE-558

A network tour of data science

Frossard Pascal, Vandergheynst Pierre

Cursus	Sem.	Type
Data Science	MA1, MA3	Obl.
Génie électrique et électronique	MA1, MA3	Obl.
Génie électrique		Obl.
Managmt, tech et entr.	MA1, MA3	Obl.
Microtechnique	MA1, MA3	Obl.
Mineur en Data science	H	Opt.

Language	English
Credits	4
Session	Winter
Semester	Fall
Exam	During the semester
Workload	120h
Weeks	14
Hours	**4 weekly**
Lecture	2 weekly
Exercises	2 weekly

Content

Context
In the last decade, our information society has mutated into a data society, where the volume of worldwide data doubles every 1.5 years. How to make sense of such tremendous volume of data? Developing effective techniques to extract meaningful information from large-scale and high-dimensional dataset has become essential for the success of business, government and science.

Objective
The goal of this course is to provide a broad introduction to effective algorithms in data science and network analysis. A major effort will be given to show that existing data analysis techniques can be defined and enhanced on graphs. Graphs encode complex structures like cerebral connection, stock exchange, and social network. Strong mathematical tools have been developed based on linear and non-linear graph spectral harmonic analysis to advance the standard data analysis algorithms. Main topics of the course are networks, unsupervised and supervised learning, recommendation, visualization, sparse representation, multi-resolution analysis, neuron network, and large-scale computing.

Structure
The course is organized into two parts: lectures (2 hours) and coding exercises (1 hour). The essential objective of the exercises is to apply the theory on real-world cases.
Evaluation
Evaluation will be conducted on a continuous basis: homeworks and coding assignments.

Keywords

data science, data mining, network science, machine learning

CS-450 Advanced algorithms

Svensson Ola Nils Anders

Cursus	Sem.	Type
Cyber security minor	E	Opt.
Data Science	MA2, MA4	Obl.
Informatique et communications		Obl.
Informatique	MA2	Obl.
Mineur en Data science	E	Opt.
Mineur en Informatique	E	Opt.
SC master EPFL	MA2, MA4	Opt.
Science et ing. computationelles	MA2, MA4	Opt.

Language	English
Credits	7
Session	Summer
Semester	Spring
Exam	Written
Workload	210h
Weeks	14
Hours	**7 weekly**
Lecture	4 weekly
Exercises	2 weekly
Project	1 weekly

Summary

A first graduate course in algorithms, this course assumes minimal background, but moves rapidly. The objective is to learn the main techniques of algorithm analysis and design, while building a repertory of basic algorithmic solutions to problems in many domains.

Content

Algorithm analysis techniques: worst-case and amortized, average-case, randomized, competitive, approximation. Basic algorithm design techniques: greedy, iterative, incremental, divide-and-conquer, dynamic programming, randomization, linear programming. Examples from graph theory, linear algebra, geometry, operations research, and finance.

Keywords

See content.

Learning Prerequisites

Required courses

An undergraduate course in Discrete Structures / Discrete Mathematics, covering formal notation (sets, propositional logic, quantifiers), proof methods (derivation, contradiction, induction), enumeration of choices and other basic combinatorial techniques, graphs and simple results on graphs (cycles, paths, spanning trees, cliques, coloring, etc.).

Recommended courses

An undergraduate course in Data Structures and Algorithms.
An undergraduate course in Probability and Statistics.

Important concepts to start the course

Basic data structures (arrays, lists, stacks, queues,trees) and algorithms (binary search; sorting; graph connectivity); basic discrete mathematics (proof methods, induction, enumeration and counting, graphs); elementary probability and statistics (random variables, distributions, independence, conditional probabilities); data abstraction.

Learning Outcomes

By the end of the course, the student must be able to:

- Use a suitable analysis method for any given algorithm
- Prove correctness and running-time bounds
- Design new algorithms for variations of problems studied in class

- Select appropriately an algorithmic paradigm for the problem at hand
- Define formally an algorithmic problem

Teaching methods

Ex cathedra lecture, reading

Assessment methods

Supervision

Office hours	Yes
Assistants	Yes
Forum	Yes
Others	For details, see the course web page.

Resources

Bibliography

See web page for the course.

Ressources en bibliothèque

- Randomized Algorithms / Motwani
- Approximation Algorithms / Vazirani
- Computational Complexity / Papadimitrou
- Algebraic Complexity Theory / Buergisser
- Quantum Computation and Quantum Information / Nielsen

Notes/Handbook

Class notes and references for the running semester will be provided as needed within a few days after each lecture.

Websites

- http://theory.epfl.ch/courses/AdvAlg/

COM-501

Advanced cryptography

Vaudenay Serge

Cursus	Sem.	Type
Cyber security minor	E	Opt.
Data Science	MA2, MA4	Opt.
Mineur en Data science	E	Opt.
SC master EPFL	MA2, MA4	Opt.

Language	English
Credits	4
Session	Summer
Semester	Spring
Exam	Written
Workload	120h
Weeks	14
Hours	**4 weekly**
Lecture	2 weekly
Exercises	2 weekly

Summary

This course reviews some failure cases in public-key cryptography. It introduces some cryptanalysis techniques. It also presents fundamentals in cryptography such as interactive proofs. Finally, it presents some techniques to validate the security of cryptographic primitives.

Content

1. **Public-key cryptography:** Factoring, RSA problem, discrete logarithm problem, attacks based on subgroups
2. **Conventional cryptography:** differential and linear cryptanalysis, hypothesis testing, decorrelation
3. **Interactive proofs:** NP-completeness, interactive systems, zero-knowledge
4. **Proofs techniques:** Security of encryption, random oracles, game reduction techniques

Keywords

cryptography, cryptanalysis, interactive proof, security proof

Learning Prerequisites

Required courses

- Cryptography and security (COM-401)

Important concepts to start the course

- Cryptography
- Mathematical reasoning
- Number theory and probability theory
- Algorithmics
- Complexity

Learning Outcomes

By the end of the course, the student must be able to:

- Assess / Evaluate the security deployed by cryptographic schemes
- Prove or disprove security

- Justify the elements of cryptographic schemes
- Analyze cryptographic schemes
- Implement attack methods
- Model security notions

Teaching methods

ex-cathedra

Expected student activities

- active participation during the course
- take notes during the course
- do the exercises during the exercise sessions
- complete the regular tests and homework
- read the material from the course
- self-train using the provided material
- do the midterm exam and final exam

Assessment methods

Mandatory continuous evaluation:

- homework (30%)
- regular graded tests (30%)
- midterm exam (40%)

Final exam averaged (same weight) with the contiuous evaluation, but with final grade between final_exam-1 and final_exam+1.

Supervision

Office hours	No
Assistants	Yes
Forum	No
Others	Lecturers and assistants are available upon appointment.

Resources

Bibliography

- Communication security: an introduction to cryptography. Serge Vaudenay. Springer 2004.
- A computational introduction to number theory and algebra. Victor Shoup. Cambridge University Press 2005.
- Algorithmic cryptanalysis. Antoine Joux. CRC 2009.

Ressources en bibliothèque

- Algorithmic cryptanalysis / Joux
- A computational introduction to number theory and algebra / Shoup
- Communication security / Vaudenay

Websites

- http://lasec.epfl.ch/teaching.shtml

COM-417 **Advanced probability and applications**

Lévêque Olivier

Cursus	Sem.	Type
Data Science	MA2, MA4	Opt.
Informatique et communications		Obl.
Informatique	MA2	Opt.
Mineur en Data science	E	Opt.
SC master EPFL	MA2, MA4	Obl.

Language	English
Credits	6
Session	Summer
Semester	Spring
Exam	Written
Workload	180h
Weeks	14
Hours	**5 weekly**
Lecture	3 weekly
Exercises	2 weekly

Summary

In this course, various aspects of probability theory are considered. The first part is devoted to the main theorems in the field (law of large numbers, central limit theorems), while the second part focuses on the theory of martingales in discrete time.

Content

I. Probability
- sigma-fields, probability measures, random variables
- independence, expectation
- convergence of sequences of random variables
- laws of large numbers- central limit theorem
- concentration inequalities
- moments

II. Martingales
- conditional expectation
- definition and properties of a martingale
- stopping times, optional stopping theorem
- maximal inequalities
- convergence theorems

Keywords

probability, measure theory, martingales, convergence theorems

Learning Prerequisites

Required courses

Basic probability course
Calculus courses

Recommended courses

complex analysis

Important concepts to start the course

This course is NOT an introductory course on probability: the students should have a good understanding and practice of basic probability concepts such as: distribution, expectation, variance, independence, conditional probability.
The students should also be at ease with calculus. Complex analyisis is a plus, but is not required.
On the other hand, no prior background on measure theory is needed for this course: we will go through the basic concepts one by one at the beginning.

Learning Outcomes

By the end of the course, the student must be able to:

- Understand the foundations of probability theory
- Acquire a solid knowledge of martingale theory

Teaching methods

Ex cathedra + exercises

Expected student activities

active participation to exercise sessions

Assessment methods

Midterm 10%, homeworks 10%, exam 80%

Resources

Bibliography

Sheldon M. Ross, Erol A. Pekoz, A Second Course in Probability,1st edition, www.ProbabilityBookstore.com, 2007.
Jeffrey S. Rosenthal, A First Look at Rigorous Probability Theory,2nd edition, World Scientific, 2006.
Geoffrey R. Grimmett, David R. Stirzaker, Probability and Random Processes,3rd edition, Oxford University Press, 2001.
Richard Durrett, Probability: Theory and Examples,4th edition, Cambridge University Press, 2010.

Ressources en bibliothèque

- A Second Course in Probability / Ross
- A First Look at Rigorous Probability Theory / Rosenthal
- Probability and Random Processes / Grimmett
- Probability: Theory and Examples / Durrett

Notes/Handbook

available on the course website

Websites

- http://ipgold.epfl.ch/~leveque/Advanced_Prob/

Prerequisite for

Advanced classes requiring a good knowledge of probability

CS-523

Advanced topics on privacy enhancing technologies

González Troncoso Carmela, Hubaux Jean-Pierre

Cursus	Sem.	Type
Data Science	MA1, MA3	Opt.
Informatique	MA1, MA3	Opt.
SC master EPFL	MA1, MA3	Opt.

Language	English
Credits	7
Session	Winter
Semester	Fall
Exam	Written
Workload	210h
Weeks	14
Hours	**6 weekly**
Lecture	3 weekly
Project	3 weekly

Summary

This course will provide students with the knowledge to tackle the design of privacy-preserving ICT systems. Students will learn about existing technologies to protect privacy, and how to evaluate the protection they provide.

Content

The course will delve into the following topics:

- Privacy definitions and concepts, and the socioeconomic context of privacy: economics and incentives, ethics, regulation.
- Cryptographic privacy solutions: Identity management and anonymous credentials, zero-knowledge proofs, secure multi-party computation, homomorphic encryption, garbled circuits, Private information retrieval (PIR), Oblivious RAM (ORAM)
- Anonymization and data hiding: k-anonymity, l-diversity, t-proximity; dummy use, differential privacy and Laplacian noise; composability
- Machine learning and privacy: how machine learning can be use to infer private information; and how much information can be learned from machine learning models.
- Protection of metadata: anonymous communications systems, location privacy, censorpship resistance.
- Online tracking and massive surveillance.
- Evaluation of privacy preserving systems - notions, definitions, quantification / computation
- Fairness and transparency and their interplay with privacy

Keywords

Privacy, anonymity, homomorphic encryption, ethics

Learning Prerequisites

Recommended courses

COM 402 Information Security and Privacy
COM 301 Computer Security

Important concepts to start the course

Basic programming skills; basics of probabilities and statistics; basics of cryptography

Learning Outcomes

By the end of the course, the student must be able to:

- Select appropriately privacy mechanisms
- Develop privacy technologies
- Assess / Evaluate privacy protection
- Reason about privacy concerns

Teaching methods

Lectures and mini-projects developing privacy enhancing technologies supervised by assistants.

Expected student activities

Attending lectures
Execute mini-projects

Assessment methods

Written final exam and mini-project during the course

Supervision

Assistants	Yes

CS-435 **Analytic algorithms**

Vishnoi Nisheeth

Cursus	Sem.	Type
Data Science	MA2, MA4	Opt.
Informatique	MA2	Opt.
SC master EPFL	MA2, MA4	Opt.

Language	English
Credits	4
Session	Summer
Semester	Spring
Exam	During the semester
Workload	120h
Weeks	14
Hours	**3 weekly**
Lecture	2 weekly
Exercises	1 weekly

Summary

In the last decade, many fundamental algorithmic problems have benefited from viewing the underlying discrete problems through the lens of continuous/analytic methods. In this course we will introduce a selection of such techniques and explore their applications.

Content

Convexity and gradient descent
Multiplicative weight update (MWU) method and online convex optimization
Gradient descent based methods for solving linear equations
Optimization problems involving polynomials
Graphs and their eigenvectors and eigenvalues
Graphs as electrical networks
Graphs Laplacians and solving Laplacian equations
Application: Fast algorithms to compute network flows (using MWU, electrical flows and Laplacian solvers)
Application: Fast algorithms for graph cuts (using eigenvectors and Laplacian solvers)
Application: Algorithms for counting perfect matchings in graphs (using convex programs involving polynomials)

Keywords

Convex optimization, Spectral methods, Polynomials, Discrete Optimization, Continuous Optimization

Learning Prerequisites

Required courses

Calculus (MATH105), Linear Algebra (MATH111), Algorithms (CS250), Theory of Computation (CS251) or equivalents, Advanced Algorithms (CS-450) (or equivalent).

Recommended courses

Important concepts to start the course

This is an advanced course and requires mathematical maturity including linear algebra, multi-variate calculus, analysis, probability and algorithms.

Learning Outcomes

By the end of the course, the student must be able to:

- Learn fundamental techniques which apply continuous methods to discrete problems
- Apply analytic techniques to a variety of related problems
- Read, understand, and explain state of the art papers in this area

Assessment methods

Homeworks, Scribe Notes, Exam and Project/Presentation*.
*Tentative

Resources

Bibliography

Books relevant to the course:
Vishnoi - Lx=b
Nesterov - Introductory lectures on convex optimization
Shalev-Schwartz - Online learning and online convex optimization
References for Basics:
Apostol - Calculus I and II
Strang - Linear algebra and its applications
Boyd and Vanderberghe - Convex optimization
Strogatz - Nonlinear dynamics and Chaos

Ressources en bibliothèque

- Lx=b / Vishnoi (print)
- Lx=b / Vishnoi (online)
- Introductory lectures on convex optimization / Nesterov
- Online learning and online convex optimization / Shalev-Schwartz
- Nonlinear dynamics and Chaos / Strogatz
- Calculus II / Apostol
- Linear algebra and its applications / Strang
- Convex optimization / Boyd
- Calculus I / Apostol

Notes/Handbook

Vishnoi - Zeros of Polynomials and their Applications to theory. Available from
http://theory.epfl.ch/vishnoi/Publications_files/ZerosIntro.pdf

Vishnoi -- A mini-course on convex optimization. Available from
http://theory.epfl.ch/vishnoi/Nisheeth-VishnoiFall2014-ConvexOptimization.pdf

MATH-493

Applied biostatistics

Goldstein Darlene

Cursus	Sem.	Type
Bioingénierie	MA2, MA4	Opt.
Data Science	MA2, MA4	Opt.
Informatique	MA2	Opt.
Ing.-math	MA2, MA4	Opt.
Mathématicien	MA2	Opt.
Mineur en Data science	E	Opt.
SC master EPFL	MA2, MA4	Opt.
Sciences du vivant	MA2, MA4	Opt.
Sciences et technologies du vivant	MA2	Opt.

Language	English
Credits	5
Session	Summer
Semester	Spring
Exam	During the semester
Workload	150h
Weeks	14
Hours	**4 weekly**
Lecture	2 weekly
Exercises	2 weekly

Summary

This course covers topics in applied biostatistics, with an emphasis on practical aspects of data analysis using R statistical software. Topics include types of studies and their design and analysis, high dimensional data analysis (genetic/genomic) and other topics as time and interest permit.

Content

- Types of studies
- Design and analysis of studies
- R statistical software
- Reproducible research techniques and tools
- Report writing
- Exploratory data analysis
- Liniear modeling (regression, anova)
- Generalized linear modeling (logistic, Poission)
- Survival analysis
- Discrete data analysis
- Meta-analysis
- High dimensional data analysis (genetics/genomics applications)
- Additional topics as time and interest permit

Keywords

Data analysis, reproducible research, statistical methods, R, biostatistical data analysis, statistical data analysis

Learning Prerequisites

Required courses

This course will be very difficult for students with no previous course or experience with statistics. **Previous experience with R is neither assumed nor required.**

Recommended courses

Undergraduate statistics course

Important concepts to start the course

It is useful to review statistical hypothesis testing.

Learning Outcomes

By the end of the course, the student must be able to:

- Interpret analysis results
- Justify analysis plan
- Plan analysis for a given dataset
- Analyze various types of biostatistical data
- Synthesize analysis into a written report
- Report plan of analysis and results obtained

Transversal skills

- Write a scientific or technical report.
- Assess one's own level of skill acquisition, and plan their on-going learning goals.
- Take feedback (critique) and respond in an appropriate manner.
- Use a work methodology appropriate to the task.

Teaching methods

Lectures and practical exercises using R. Typically, each week covers an analysis method in the lecture and then the corresponding exercise session consists of an R proactical showing how to implement the methods using R. In each practical, students use R to carry out analyses of the relevant data type for that week.

Expected student activities

Students are expected to participate in their learning by attending lectures and practical exercise sessions, posing questions, proposing topics of interest, peer reviewing of preliminary reports, and interacting with teaching staff regarding their understanding of course material. In addition, there will be a number of short activities in class aimed at improving English for report writing.

Assessment methods

Evaluation is based on written reports of projects analyzing biostatistical data.

CS-401

Applied data analysis

West Robert

Cursus	Sem.	Type
Bioengineering	MA1, MA3	Opt.
Computational Neurosciences minor	H	Opt.
Computational science and Engineering	MA1, MA3	Opt.
Computer science	MA1, MA3	Opt.
Data Science	MA1, MA3	Obl.
Data science minor	H	Opt.
Digital Humanities	MA1, MA3	Obl.
Electrical Engineering		Obl.
Electrical and Electronical Engineering	MA1, MA3	Opt.
Financial engineering	MA1, MA3	Opt.
Internet of Things minor	H	Opt.
Life sciences and technologies	MA1	Opt.
Managmt, tech et entr.	MA1, MA3	Opt.
SC master EPFL	MA1, MA3	Opt.

Language	English
Credits	6
Session	Winter
Semester	Fall
Exam	Written
Workload	180h
Weeks	14
Hours	**4 weekly**
Lecture	2 weekly
Project	2 weekly

Summary

This course teaches the basic techniques and practical skills required to make sense out of a variety of data, with the help of the most acclaimed software tools in the data science world: pandas, scikit-learn, Spark, etc.

Content

Thanks to a new breed of software tools that allows to easily process and analyze data at scale, we are now able to extract invaluable insights from the vast amount of data generated daily. As a result, both the business and scientific world are undergoing a revolution which is fueled by one of the most sought after job profiles: the data scientist.
This course covers the fundamental steps of the data science pipeline:

Data Acquisition

- Variety as one of the main challenges in big data: structured, semi-structured, unstructured
- Data sources: open, public (scraping, parsing, and down-sampling)
- Dataset fusion, filtering, slicing & dicing
- Data granularities and aggregations

Data Wrangling

- Data manipulation, array programming, dataframes
- The many sources of data problems (and how to fix them): missing data, incorrect data, inconsistent representations
- Schema alignment, data reconciliation
- Data quality testing with crowdsourcing

Data Interpretation

- Stats in practice (distribution fitting, statistical significance, etc.)
- Co-occurrence grouping (market-basket analysis)
- Machine learning in practice (supervised and unsupervised, feature engineering, more data vs. advanced algorithms, curse of dimensionality, etc.)
- Text mining: vector space model, topic models, word embedding
- Social network analysis (influencers, community detection, etc.)

Data Visualization

- Introduction to different plot types (1, 2, and 3 variables), layout best practices, network and geographical data
- Visualization to diagnose data problems, scaling visualization to large datasets, visualizing uncertain data

Reporting

- Results reporting, infographics
- How to publish reproducible results
- Anonymiziation, ethical concerns

The students will learn the techniques during the ex-cathedra lectures, and will then get familiar with the software tools to complete the homework assignments (which will be in part executed under the supervision of the teacher and the assistants, during the lab hours).
In parallel, the students will embark in a semester-long project, split in agile teams of 3. The outcome of such team efforts will be unified towards the end of the course, to build a project portfolio that will be made public (and available as open-source).
At the end of the semester, students will also take a 3-hour final exam in a classroom with computers, where they will be asked to complete a data analysis pipeline (both with code and extensive comments) on a dataset they have never worked with before.

Keywords

data science, data analysis, data mining, machine learning

Learning Prerequisites

Required courses

The student MUST have passed an introduction to databases course, OR a course in probability & statistics, OR two separate courses that include programming projects.

Recommended courses

- CS-423 Distributed Information Systems
- CS-433 Machine Learning

Important concepts to start the course

Algorithms, object oriented programming, basic probability and statistics

Learning Outcomes

By the end of the course, the student must be able to:

- Construct a coherent understanding of the techniques and software tools required to perform the fundamental steps of the Data Science pipeline
- Perform data acquisition (data formats, dataset fusion, Web scrapers, Rest APIs, Open Data, Big Data platforms, etc.)
- Perform data wrangling (fixing missing and incorrect data, data reconciliation, data quality assessments, etc.)
- Perform data interpretation (statistics, knowledge extraction, critical thinking, team discussions, ad-hoc visualizations, etc.)

- Perform result dissemination (reporting, visualizations, publishing reproducible results, ethical concerns, etc.)
- Perform data acquisition (data formats, dataset fusion, Web scrapers, REST APIs, open data, big data platforms, etc.)

Transversal skills

- Give feedback (critique) in an appropriate fashion.
- Demonstrate the capacity for critical thinking
- Write a scientific or technical report.
- Evaluate one's own performance in the team, receive and respond appropriately to feedback.

Teaching methods

- Physical in-class recitations and lab sessions
- Homework assignments
- Course project

Expected student activities

Students are expected to:

- Attend the lectures and lab sessions
- Complete a weekly homework assignment
- Read/watch the pertinent material before a lecture
- Engage during the class, and present their results in front of the other colleagues

Assessment methods

- 30% continuous assessment during the semester (homework)
- 30% final exam, data analysis task on a computer (3 hours)
- 40% final project, done in groups of 3

Supervision

Office hours	Yes
Assistants	Yes
Forum	Yes
Others	http://ada.epfl.ch

Resources

Virtual desktop infrastructure (VDI)

No

Websites

- http://ada.epfl.ch

CS-456 **Artificial neural networks**

Gerstner Wulfram

Cursus	Sem.	Type
Biocomputing minor	E	Opt.
Computational science and Engineering	MA2, MA4	Opt.
Computer science	MA2	Opt.
Data Science	MA2, MA4	Opt.
SC master EPFL	MA2, MA4	Opt.

Language	English
Credits	4
Session	Summer
Semester	Spring
Exam	Written
Workload	120h
Weeks	14
Hours	**3 weekly**
Lecture	2 weekly
Exercises	1 weekly

Summary

Since 2010 approaches in deep learning have revolutionized fields as diverse as computer vision, machine learning, or artificial intelligence. This course gives a systematic introduction into the main models of deep artificial neural networks: Supervised Learning and Reinforcement Learning.

Content

- *Simple perceptrons for classification*
- *BackProp and Multilayer Perceptrons for deep learning*
- *Statistical Classification by deep networks*
- *Regularization and Tricks of the Trade in deep learning*
- *Error landscape and optimization methods for deep networks*
- *Convolutional networks*
- *Sequence prediction and recurrent networks*
- *Reinforcement Learning 1: Bellman equation and SARSA*
- *Reinforcement Learning 2: variants of SARSA, Q-learning, n-step-TD learning*
- *Reinforcement Learning 3: Policy gradient*
- *Deep reinforcement learning: applications*
- *Reinforcement learning and the brain*

Keywords

Deep learning, artificial neural networks, reinforcement learning, TD learning, SARSA,

Learning Prerequisites

Required courses

CS 433 Pattern Classification and Machine Learning (or equivalent)
Calculus, Linear Algebra (at the level equivalent to first 2 years of EPFL in STI or IC, such as Computer Science, Physics or Electrical Engineering)

Recommended courses

stochastic processes
optimization

Important concepts to start the course

- *Regularization in machine learning,*
- *Training base versus Test base, cross validation.*
- *Gradient descent. Stochastic gradient descent.*
- *Expectation, Poisson Process, Bernoulli Process.*

Learning Outcomes

By the end of the course, the student must be able to:

- Apply learning in deep networks to real data
- Assess / Evaluate performance of learning algorithms
- Elaborate relations between different mathematical concepts of learning
- Judge limitations of algorithms
- Propose algorithms and models for learning in deep networks

Transversal skills

- Continue to work through difficulties or initial failure to find optimal solutions.
- Manage priorities.
- Access and evaluate appropriate sources of information.
- Write a scientific or technical report.

Teaching methods

ex cathedra lectures and miniproject

Expected student activities

work on miniproject
solve all exercises
attend all lectures and take notes during lecture, participate in quizzes.
If you cannot attend a lecture, then you must read the recommended book chapters

Assessment methods

written exam (70 percent) and miniproject (30 percent)

Supervision

Office hours	No
Assistants	Yes
Forum	Yes
Others	TAs are available during exercise sessions. Professor is available during the breaks of class. Some of the exercises are run as 'integrated exercises' during the lecture

Resources

Bibliography

- Textbook: Deep Learning by Goodfellow, Bengio, Courville (MIT Press)
- Textbook: Reinforcement Learning by Sutton and Barto (MIT Press)

Pdfs of the preprint version for both books are availble online

Ressources en bibliothèque

- Deep Learning / Goodfellow

Moodle Link

- https://moodle.epfl.ch/enrol/index.php?id=15633

COM-415

Audio and acoustic signal processing

Faller Christof, Kolundzija Mihailo

Cursus	Sem.	Type
Computer science	MA1, MA3	Opt.
Data Science	MA1, MA3	Opt.
Data science minor	H	Opt.
Digital Humanities	MA1, MA3	Opt.
SC master EPFL	MA1, MA3	Opt.

Language	English
Credits	5
Session	Winter
Semester	Fall
Exam	Written
Workload	150h
Weeks	14
Hours	**5 weekly**
Lecture	2 weekly
Exercises	2 weekly
Practical work	1 weekly

Summary

The objective of the course is to introduce theory, methods, ans basic psychoacoustics that is needed to understand state-of-the-art techniques used in pro audio and consumer audio, including microphones, surround sound, mixing and audio coding.

Content

• Acoustics and audio is covered and the manipulation and processing of audio signals. It is shown how Fourier analysis of a sound field yields the representation of the sound field with plane waves. These and other acoustic insights are used to explain microphone techniques and reproduction of sound fields.

• Psychoacoustics, loudness perception and spatial hearing are covered in detail. The latter is used to motivate stereo and surround mixing and audio playback. Audio playback is put into contex with a detailed coverage of room acoustics.

• The short-time Fourier transform is introduced as a tool for flexible manipulation of audio signals, such as filtering, delaying and other spectral modification. Matrix surround, audio coding, and beamforming are also treated.

Learning Prerequisites

Recommended courses

Signal processing for communication, any course on Signals and Systems

Learning Outcomes

By the end of the course, the student must be able to:

• Apply basics of acoustics, signal processing, reproduction and capture

• Understand and implement linear and adaptative filtering, beamforming, noise suppression, audio coding, stereo and multichannel sound capture and reproduction

Teaching methods

In class ex-cathedra + exercices + mini-project supervision

Expected student activities

• Theoretical and practical exercises

• Mini-projects : individual or in small groups

Assessment methods

- Final exam
- Midterm exam
- Mini-project

Supervision

Office hours	Yes
Assistants	Yes
Forum	Yes

EE-554

Automatic speech processing

Bourlard Hervé

Cursus	Sem.	Type
Computer science	MA1, MA3	Opt.
Data Science	MA1, MA3	Opt.
Digital Humanities	MA1, MA3	Opt.
Electrical and Electronical Engineering	MA1, MA3	Opt.
SC master EPFL	MA1, MA3	Opt.

Language	English
Credits	3
Session	Winter
Semester	Fall
Exam	Written
Workload	90h
Weeks	14
Hours	**3 weekly**
Lecture	2 weekly
Exercises	1 weekly

Summary

The goal of this course is to provide the students with the main formalisms, models and algorithms required for the implementation of advanced speech processing applications (involving, among others, speech coding, speech analysis/synthesis, and speech recognition).

Content

1. Introduction: Speech processing tasks, language engineering applications.

2. Basic Tools: Analysis and spectral properties of the speech signal, linear prediction algorithms, statistical pattern recognition, dynamic programming.

3. Speech Coding: Human hearing properties, quantization theory, speech coding in the temporal and frequency domains.

4. Speech Synthesis: Morpho-syntactic analysis, phonetic transcription, prosody, speech synthesis models.

5. Automatic Speech Recognition: Temporal pattern matching and Dynamic Time Warping (DTW) algorithms, speech recognition systems based on Hidden Markov Models (HMMs).

6. Speaker recognition and speaker verification: Formalism, hypothesis testing, HMM based speaker verification.

7. Linguistic Engineering: state-of-the-art and typical applications

Keywords

speech processing, speech coding, speech analysis/synthesis, automatic speech recognition, speaker identification, text-to-speech

Learning Prerequisites

Required courses

Basis in linear algebra, signal processing (FFT), and statistics

Important concepts to start the course

Basic knowledge in signal processing, linear algebra, statistics and stochastic processes.

Learning Outcomes

By the end of the course, the student must be able to:

- speech signal properties
- Exploit those properties to speech codign, speech synthesis, and speech recognition

Transversal skills

- Use a work methodology appropriate to the task.
- Access and evaluate appropriate sources of information.
- Use both general and domain specific IT resources and tools

Teaching methods

Lecture + lab exercises

Expected student activities

Attending courses and lab exercises. Read additional papers and continue lab exercises at home if necessary. Regulary answer list of questions for feedback.

Assessment methods

Written exam without notes

Resources

Ressources en bibliothèque

- Traitement de la parole / Boite

Websites

- http://lectures.idiap.ch/

BIO-465 **Biological modeling of neural networks**

Gerstner Wulfram

Cursus	Sem.	Type
Auditeurs en ligne	E	Obl.
Biocomputing minor	E	Opt.
Biomedical technologies minor	E	Opt.
Computational Neurosciences minor	E	Opt.
Computational science and Engineering	MA2, MA4	Opt.
Computer science	MA2	Opt.
Data Science	MA2, MA4	Opt.
Electrical and Electronical Engineering	MA2, MA4	Opt.
Life sciences and technologies	MA2	Opt.
Neuroprosthetics minor	E	Opt.
SC master EPFL	MA2, MA4	Opt.
Sciences du vivant	MA2, MA4	Opt.

Language	English
Credits	4
Session	Summer
Semester	Spring
Exam	Written
Workload	120h
Weeks	14
Hours	**4 weekly**
Lecture	2 weekly
Exercises	2 weekly

Summary

In this course we study mathematical models of neurons and neuronal networks in the context of biology and establish links to models of cognition.

Content

I. Models of single neurons 1. Introduction: brain vs computer and a first simple neuron model 2. Models on the level of ion current (Hodgkin-Huxley model) 3./4. Two-dimensional models and phase space analysis ***II. Neuronal Dynamics of Cognition*** 5./6. Associative Memory and Attractor Dynamics (Hopfield Model) 7. Neuronal Populations and networks 8. Continuum models and perception 9. Competition and models of Decision making ***III. Noise and the neural code*** 10. Noise and variability of spike trains (point processes, renewal process, interval distribution) 11: Variance of membrane potentials and Spike Response Models **IV. Plasticity and Learning** 12. Synaptic Plasticity and Long-term potentiation and Learning (Hebb rule, mathematical formulation) 13. Summary: Fitting Neural Models to Data

Keywords

neural networks, neuronal dynamics, computational neuroscience, mathematical modeling in biology, applied mathematics, brain, cognition, neurons, memory, learning, plasticity

Learning Prerequisites

Required courses

undergraduate math at the level of electrical engineering or physics majors
undergraduate physics.

Recommended courses

Analysis I-III, linear algebra, probability and statistics
For SSV students: Dynamical Systems Theory for Engineers or "Mathematical and Computational Models in Biology"

Important concepts to start the course

Differential equations, stochastic processes,

Learning Outcomes

By the end of the course, the student must be able to:

- Analyze two-dimensional models in the phase plane
- Solve linear one-dimensional differential equations
- Develop a simplified model by separation of time scales
- Analyze connected networks in the mean-field limit
- Formulate stochastic models of biological phenomena
- Formalize biological facts into mathematical models
- Prove stability and convergence
- Apply model concepts in simulations
- Predict outcome of dynamics
- Describe neuronal phenomena

Transversal skills

- Plan and carry out activities in a way which makes optimal use of available time and other resources.
- Collect data.
- Write a scientific or technical report.

Teaching methods

Classroom teaching, exercises and miniproject

Expected student activities

- participate in ALL in-class exercises.
- do all homework exercises (paper-and-pencil)
- study video lectures if you miss a class
- study suggested textbook sections for in-depth understanding of material
- submit miniprojects

Assessment methods

Written exam (67%) & miniproject (33%)

Supervision

Office hours	No
Assistants	Yes
Forum	Yes
Others	The teacher is available during the breaks of the class. Some exercises are integrated in class in the presence of the teacher and the teaching assistants.

Resources

Bibliography

Gerstner, Kistler, Naud, Pansinski : Neuronal Dynamics, Cambridge Univ. Press 2014

Ressources en bibliothèque

- Neuronal Dynamics / Gerstner

Notes/Handbook

The textbook is online at: http://neuronaldynamics.epfl.ch/

Videos

- http://lcn.epfl.ch/~gerstner/NeuronalDynamics-MOOC1.html
- http://lcn.epfl.ch/~gerstner/NeuronalDynamics-MOOC2.html

MATH-460 **Combinatorial optimization**

Cursus	Sem.	Type
Data Science	MA1, MA3	Opt.
Electrical Engineering		Obl.
Ing.-math	MA1, MA3	Opt.
Mathématicien	MA1, MA3	Opt.

Language	English
Credits	5
Session	Winter
Semester	Fall
Exam	Written
Workload	150h
Weeks	14
Hours	**4 weekly**
Lecture	2 weekly
Exercises	2 weekly

Remarque

pas donné en 2018-19

Summary

The guiding question of Combinatorial Optimization is: How do I efficiently select an optimal solution among a finite but very large set of alternatives? We will address the solution of this question in the context of classical discrete optimization problems.

Content

- Paths and flows: Stronlgly polynomial time algorithms for shortest paths and minimum cost network flows
- Minimum spanning trees and matroids: Greedy, Kruskal's and Prim's algorithm
- Arborescences and matroid intersection
- Polyhedra and approximation algorithms
- Maximum weight matchings in general graphs and the matching polytope

Keywords

- Algorithm
- Polyhedron
- Matroid
- NP-completeness

Learning Prerequisites

Required courses

Discrete optimization (Second year math.)

Learning Outcomes

By the end of the course, the student must be able to:

- Choose an appropriate method for solving a combinatorial optimization problem
- Prove theorems in discrete optimization
- Design algorithms
- Analyze efficiency of algorithms

Transversal skills

- Demonstrate a capacity for creativity.
- Continue to work through difficulties or initial failure to find optimal solutions.
- Assess one's own level of skill acquisition, and plan their on-going learning goals.

Teaching methods

Ex cathedra lecture and exercises to be solved at home and in the classroom

Expected student activities

Attendance of lectures and exercises
Completion of exercises at home
Study of literature

Assessment methods

Written exam during exam session

Supervision

Office hours	Yes
Assistants	Yes
Forum	No

Resources

Bibliography

Alexander Schrijver, Combinatorial Optimization: Polyhedra and Efficiency, Springer-Verlag.

CS-524 **Computational complexity**

Svensson Ola Nils Anders

Cursus	Sem.	Type
Computer science	MA1, MA3	Opt.
Data Science	MA1, MA3	Opt.
SC master EPFL	MA1, MA3	Opt.

Language	English
Credits	4
Session	Winter
Semester	Fall
Exam	During the semester
Workload	120h
Weeks	14
Hours	**4 weekly**
Lecture	3 weekly
Exercises	1 weekly

Summary

In computational complexity we study the computational resources needed to solve problems and understand the relation between different types of computation. This course advances the students knowledge of computational complexity, and develop an understanding of fundamental open questions.

Content

- Complexity classes (time, space, nondeterminism)
- Boolean circuits and nonuniform computation
- Role of randomness in computation (extractors, pseudo-random generators)
- Interactive proofs and zero knowledge proofs
- Probabilistically checkable proofs and their characterization of the complexity class NP (PCP Theorem)
- Communication complexity

Keywords

theoretical computer science
computational complexity

Learning Prerequisites

Recommended courses

Theory of computation (CS-251)
Algorithms (CS-250)

Learning Outcomes

By the end of the course, the student must be able to:

- Demonstrate an understanding of computational complexity and the P vs NP problem
- Formalize and analyze abstractions of complex scenarios/problems
- Express a good understanding of different concepts of proofs
- Prove statements that are similar to those taught in the course
- Use and understand the role of randomness in computation
- Illustrate a basic understanding of probabilistically checkable proofs and their characterization of the class NP (the PCP-Theorem)

- Explain recent exciting developments in theoretical computer science
- Compare different models of computation

Transversal skills

- Demonstrate the capacity for critical thinking
- Summarize an article or a technical report.

Teaching methods

Lecturing and exercises

Expected student activities

Actively attending lectures and exercise sessions. Also homeworks and exam.

Supervision

Office hours	Yes
Assistants	Yes
Forum	Yes

Resources

Virtual desktop infrastructure (VDI)

No

Bibliography

Sanjeev Arora and Boaz Barak: *Computational Complexity: A Modern Approach, Cambridge University Press.*

Ressources en bibliothèque

- Computational Complexity: A Modern Approach / Arora

Websites

- http://theory.epfl.ch/courses/complexity/

MATH-453

Computational linear algebra

Vacat .

Cursus	Sem.	Type
Computational science and Engineering	MA2, MA4	Opt.
Data Science	MA2, MA4	Opt.
Ing.-math	MA2, MA4	Opt.
Mathematics for teaching	MA2, MA4	Opt.
Mathématicien	MA2	Opt.

Language	English
Credits	5
Session	Summer
Semester	Spring
Exam	Oral
Workload	150h
Weeks	14
Hours	**4 weekly**
Lecture	2 weekly
Exercises	2 weekly

Summary

This course provides an overview of state-of-the-art techniques for solving large-scale linear algebra problems, as they typically arise in applications. A central goal of this course is to give the ability to choose a suitable solver for a given application.

Content

Introduction
Sources of large-scale linear algebra problems. Recap of required linear algebra concepts.
Eigenvalue problems
Krylov subspace methods. Singular value problems. Preconditioned iterative methods.
Linear systems
Direct sparse factorizations. Krylov subspace methods and preconditioners.
Matrix functions
Theory and algorithms.

Keywords

linear systems, eigenvalue problems, matrix functions

Learning Prerequisites

Required courses

Linear Algebra, Numerical Analysis

Learning Outcomes

By the end of the course, the student must be able to:

- Choose method for solving a specific problem.
- Prove the convergence of iterative methods.
- Interpret the results of a computation in the light of theory.
- Implement numerical algorithms.
- Describe methods for solving linear algebra problems.
- State theoretical properties of numerical algorithms.

Teaching methods

Ex cathedra lecture, exercises in the classroom and with computer

Expected student activities

Attendance of lectures.
Completing exercises.
Solving problems on the computer.

Assessment methods

Oral examination.
Dans le cas de l'art. 3 al. 5 du Règlement de section, l'enseignant décide de la forme de l'examen qu'il communique aux étudiants concernés.

Resources

Bibliography

Lecture notes will be provided by the instructor. Complimentary reading:
H. Elman, D. J. Silvester, and A. J. Wathen. Finite elements and fast iterative solvers: with applications in incompressible fluid dynamics. Oxford University Press, 2005.
G. H. Golub and C. Van Loan. Matrix computations. Johns Hopkins University Press, 1996.
Y. Saad. Iterative methods for sparse linear systems. Second edition. SIAM, 2003.

Ressources en bibliothèque

- Finite elements and fast iterative solvers / Elman
- Iterative methods for sparse linear systems / Saad
- Matrix computations / Golub

CS-413 Computational photography

Cursus	Sem.	Type
Computer science	MA2	Opt.
Data Science	MA2, MA4	Opt.
Digital Humanities	MA2	Opt.
Electrical and Electronical Engineering	MA2, MA4	Opt.
SC master EPFL	MA2, MA4	Opt.

Language	English
Credits	5
Session	Summer
Semester	Spring
Exam	During the semester
Workload	150h
Weeks	14
Hours	**4 weekly**
Lecture	2 weekly
Project	2 weekly

Remarque

pas donné en 2018-19

Summary

The students will gain the theoretical knowledge in computational photography, which allows recording and processing a richer visual experience than traditional digital imaging. They will also execute practical group projects to develop their own computational photography application.

Content

Computational photography is the art, science, and engineering of creating a great (still or moving) image. Information is recorded in space, time, across visible and invisible radiation and from other sources, and then post-processed to produce the final - visually pleasing - result.
Basics: Human vision system, Light and illumination, Geometric optics, Color science, Sensors, Digital camera systems.
Generalized illumination: Structured light, High dynamic range (HDR) imaging, Time-of-flight.
Generalized optics: Coded Image Sensing, Coded aperture, Focal stacks.
Generalized sensing: Low light imaging, Depth imaging, Plenoptic imaging, Light field cameras.
Generalized processing: Super-resolution, In-painting, Compositing, Photomontages, Panoramas, HDR imaging, Multi-wavelength imaging, Dynamic imaging.
Generalized display: Stereoscopic displays, HDR displays, 3D displays, Mobile displays.

Keywords

Computational Photography, Coded Image Sensing, Non-classical image capture, Multi-Image & Sensor Fusion, Mobile Imaging.

Learning Prerequisites

Required courses

- A basic Signal Processing, Image Processing, and/or Computer Vision course.
- Linear Algebra.

Recommended courses

- Introduction to Computer Vision.
- Signal Processing for Communications.

Important concepts to start the course

- Basic signal processing.
- Basic computer vision.
- Basic programming (iOS, Android, Matlab).

Learning Outcomes

By the end of the course, the student must be able to:

- Identify the main components of a computational photography system.
- Contextualise the main trends in computational optics, sensing, processing, and displays.
- Create a computational photography application on a mobile platform.
- Design a computational photography solution to solve a particular imaging task.
- Assess / Evaluate hardware and software combinations for their imaging performance.
- Formulate computational photography challenges that still need to be resolved.

Transversal skills

- Evaluate one's own performance in the team, receive and respond appropriately to feedback.
- Continue to work through difficulties or initial failure to find optimal solutions.

Teaching methods

The course consists of 2 hours of lectures per week that will cover the theoretical basics. An additional 2 hours per week are dedicated to a group project designing, developing, and programming a compuational photography application on a mobile plateform (iOS, Android).

Expected student activities

The studens is expected to attend the class and actively participate in the practical group project, which requires coding on either Android or iOS plateform. The student is also required to read the assigned reading material (book chapters, scientific articles).

Assessment methods

The theoretical part will be evaluated with an oral exam at the end of the semester, and the practical part based on the students' group projects.

Resources

Bibliography

- Selected book chapters
- Course notes (on moodle)
- Links to relevant scientific articles and on-line resources will be given on moodle.

CS-442

Computer vision

Fua Pascal

Cursus	Sem.	Type
Communication systems minor	E	Opt.
Computer science minor	E	Opt.
Computer science	MA2	Obl.
Data Science	MA2, MA4	Opt.
Data science minor	E	Opt.
Robotics	MA2	Opt.
SC master EPFL	MA2, MA4	Opt.

Language	English
Credits	4
Session	Summer
Semester	Spring
Exam	Written
Workload	120h
Weeks	14
Hours	**3 weekly**
Lecture	2 weekly
Exercises	1 weekly

Summary

Computer Vision aims at modeling the world from digital images acquired using video or infrared cameras, and other imaging sensors. We will focus on images acquired using digital cameras. We will introduce basic processing techniques and discuss their field of applicability.

Content

Introduction

- History of Computer Vision
- Human vs Machine Vision
- Image formation

Extracting 2D Features

- Contours
- Texture
- Regions

3D Shape Recovery

- From one single image
- From multiple images

Learning Prerequisites

Recommended courses

Foundations of Image Science

Learning Outcomes

By the end of the course, the student must be able to:

- Choose relevant algorithms in specific situations
- Perform simple image-understanding tasks

Teaching methods

Ex cathedra lectures and programming exercises using matlab.

Assessment methods

With continuous control

Resources

Bibliography

• R. Szeliski, Computer Vision: Computer Vision: Algorithms and Applications, 2010.
• A. Zisserman and R. Hartley, Multiple View Geometry in Computer Vision, Cambridge University Press, 2003.

Ressources en bibliothèque

• Multiple View Geometry in Computer Vision / Zisserman
• Computer Vision: Algorithms and Applications / Szeliski

Websites

• http://cvlab.epfl.ch/

Moodle Link

• http://moodle.epfl.ch/course/view.php?id=472

CS-454 **Convex optimization and applications**

Lebret Hervé

Cursus	Sem.	Type
Computational science and Engineering	MA2, MA4	Opt.
Computer science	MA2	Opt.
Data Science	MA2, MA4	Opt.
SC master EPFL	MA2, MA4	Opt.
Systems Engineering minor	E	Opt.

Language	English
Credits	4
Session	Summer
Semester	Spring
Exam	During the semester
Workload	120h
Weeks	14
Hours	**3 weekly**
Lecture	1 weekly
Exercises	2 weekly

Summary

Optimization is not only a major segment of applied mathematics, it is also a critical problem in many engineering and economic fields. In any situation where resources are limited, decision makers try to solve problems they face in the best possible manner. The course provides theory and practice.

Content

The class will cover topics such as:
Convex sets and functions
Recognizing convex optimization problems
Optimality Conditions and Duality
Linear Programming (geometry of linear programming, applications in network optimization, the simplex method)
Least squares and quadratic programs
Semidefinite programming
Interior point methods

Keywords

Convex Optimisation

Learning Prerequisites

Required courses

A good background in linear algebra. Mastering MATLAB is a plus!

Recommended courses

Basic Linear Algebra

Learning Outcomes

By the end of the course, the student must be able to:

- Solve Convex optimization problems

Teaching methods

Ex-cathedra lectures (2h) and exercise sessions (1h - corrections of previous week exercises) (in English).

Assessment methods

Midterm (25%) and final exam (50%). Small personal project (25%). Exams are open-text and on paper (no use of computers)

Supervision

Office hours	No
Assistants	No
Others	From experience, the class is in fact more 2 hours of lectures and 1 hour of corrections of exercises previous session.

Resources

Bibliography

Book : Convex Optimization by Stephen Boyd and Lieven Vandenberghe

Ressources en bibliothèque

- Convex Optimization / Boyd

Websites

- http://cvxr.com/cvx/
- http://cvxopt.org/

Moodle Link

- https://moodle.epfl.ch/enrol/index.php?id=14397

COM-401

Cryptography and security

Vaudenay Serge

Cursus	Sem.	Type
Communication systems minor	H	Opt.
Computer and Communication Sciences		Obl.
Computer science minor	H	Opt.
Computer science	MA1, MA3	Obl.
Cyber security minor	H	Opt.
Data Science	MA1, MA3	Opt.
Financial engineering	MA1, MA3	Opt.
SC master EPFL	MA1, MA3	Obl.

Language	English
Credits	7
Session	Winter
Semester	Fall
Exam	Written
Workload	210h
Weeks	14
Hours	**6 weekly**
Lecture	4 weekly
Exercises	2 weekly

Summary

This course introduces the basics of cryptography. We review several types of cryptographic primitives, when it is safe to use them and how to select the appropriate security parameters. We detail how they work and sketch how they can be implemented.

Content

1. **Ancient cryptography**: Vigenère, Enigma, Vernam cipher, Shannon theory
2. **Diffie-Hellman cryptography**: algebra, Diffie-Hellman, ElGamal
3. **RSA cryptography**: number theory, RSA, factoring
4. **Elliptic curve cryptography**: elliptic curves over a finite field, ECDH, ECIES
5. **Symmetric encryption**: block ciphers, stream ciphers, exhaustive search
6. **Integrity and authentication**: hashing, MAC, birthday paradox
7. **Applications to symmetric cryptography**: mobile telephony, Bluetooth, WiFi
8. **Public-key cryptography**: cryptosystem, digital signature
9. **Trust establishment**: secure communication, trust setups
10. **Case studies**: Bluetooth, TLS, SSH, PGP, biometric passport

Keywords

cryptography, encryption, secure communication

Learning Prerequisites

Required courses

- Algebra (MATH-310)
- Probability and statistics (MATH-310)
- Algorithms (CS-250)

Recommended courses

- Network security (COM-301)

Important concepts to start the course

- Mathematical reasoning
- Probabilities

- Algebra, arithmetics
- Algorithmics

Learning Outcomes

By the end of the course, the student must be able to:

- Choose the appropriate cryptographic primitive in a security infrastructure
- Judge the strength of existing standards
- Assess / Evaluate the security based on key length
- Implement algorithms manipulating big numbers and use number theory
- Use algebra and probability theory to analyze cryptographic algorithms
- Identify the techniques to secure the communication and establish trust

Teaching methods

ex-cathedra

Expected student activities

- active participation during the course
- take notes during the course
- do the exercises during the exercise sessions
- complete the regular tests and homework
- read the material from the course
- self-train using the provided material
- do the midterm exam and final exam

Assessment methods

Mandatory continuous evaluation:

- homework (30%)
- regular graded tests (30%)
- midterm exam (40%)

Final exam averaged (same weight) with the contiuous evaluation, but with final grade between final_exam-1 and final_exam+1.

Supervision

Office hours	No
Assistants	Yes
Forum	No
Others	Lecturers and assistants are available upon appointment.

Resources

Bibliography

- Communication security: an introduction to cryptography. Serge Vaudenay. Springer 2004.
- A computational introduction to number theory and algebra. Victor Shoup. Cambridge University Press

2005.

Ressources en bibliothèque

- A computational introduction to number theory and algebra / Shoup
- Communication security / Vaudenay

Websites

- http://lasec.epfl.ch/teaching.shtml

Prerequisite for

- Advanced cryptography (COM-401)
- Algorithms in public-key cryptography (COM-408)

COM-480

Data visualization

Benzi Kirell Maël

Cursus	Sem.	Type
Computer science	MA1, MA3	Opt.
Data Science	MA1, MA3	Opt.
Data science minor	H	Opt.
Digital Humanities	MA1, MA3	Opt.
Electrical Engineering		Obl.
Electrical and Electronical Engineering	MA1, MA3	Opt.
SC master EPFL	MA1, MA3	Opt.

Language	English
Credits	4
Session	Winter
Semester	Fall
Exam	During the semester
Workload	120h
Weeks	14
Hours	**4 weekly**
Lecture	2 weekly
Project	2 weekly

Summary

Understanding why and how to present complex data interactively in an effective manner has become a crucial skill for any data scientist. In this course, you will learn how to design, judge, build and present your own interactive data visualizations.

Content

Tentative course schedule
Week 1: Introduction to Data visualization Web development
Week 2: Javascript
Week 3: More Javascript
Week 4: Data Data driven documents (D3.js)
Week 5: Interaction, filtering, aggregation (UI /UX). Advanced D3 / javascript libs
Week 6: Perception, cognition, color Marks and channels
Week 7: Designing visualizations (UI/UX) Project introduction Dos and don’ts for data-viz
Week 8: Maps (theory) Maps (practice)
Week 9: Text visualization
Week 10: Graphs
Week 11: Tabular data viz Music viz
Week 12: Introduction to scientific visualisation
Week 13: Storytelling with data / data journalism Creative coding
Week 14: Wrap-Up

Keywords

Data viz, visualization, data science

Learning Prerequisites

Required courses

CS-305 Software engineering (BA)
CS-250 Algorithms (BA)
CS-401 Applied data analysis (MA)

Recommended courses

EE-558 A Network Tour of Data Science (MA)
CS-486 Human computer interaction (MA)
CS-210 Functional programming (BA)

Important concepts to start the course

Being autonomous is a prerequisite, we don't offer office hours and we won't have enough teaching

assistants (you've been warned!).
Knowledge of one of the following progrmaming language such as C++, Python, Scala.
Familiarity with web-development (you already have a blog, host a webiste). Experience with HTML5, Javascript is a strong plus for the course.

Learning Outcomes

By the end of the course, the student must be able to:

- Judge visualization in a critical manner and suggest improvements.
- Design and implement visualizations from the idea to the final product according to human perception and cognition
- Know the common data-viz techniques for each data domain (multivariate data, networks, texts, cartography, etc) with their technical limitations
- Create interactive visualizations int he browser using HTM5 and Javascript

Transversal skills

- Communicate effectively, being understood, including across different languages and cultures.
- Negotiate effectively within the group.
- Resolve conflicts in ways that are productive for the task and the people concerned.

Teaching methods

Ex cathedra lectures, exercises, and group projects

Expected student activities

- Follow lectures
- Read lectures notes and textbooks
- Create an advanced data-viz in groups of 3.
- Answer questions assessing the evolution of the project.
- Create a 2min screencast presentation of the viz.
- Create a process book for the final data viz.

Assessment methods

- Data-viz (35%)
- Technical implementation (15%)
- Website, presentation, screencast (15%)
- Process book (35%)

Supervision

Office hours	No
Assistants	No
Forum	No

Resources

Bibliography

Visualization Analysis and Design by Tamara Munzner, CRC Press (2014). Fee online version at EPFL.
Interactive Data Visualization for the Web by Scott Murray O'Reilly (2013) - D3 - Free online version.

Ressources en bibliothèque

- Visualization Analysis and Design / Munzner
- Interactive Data Visualization for the Web / Murray

Notes/Handbook

Lecture notes

Websites

- https://www.kirellbenzi.com

Moodle Link

- https://moodle.epfl.ch/course/view.php?id=15487

CS-411

Digital education & learning analytics

Dillenbourg Pierre, Jermann Patrick

Cursus	Sem.	Type
Computer science	MA1, MA3	Opt.
Data Science	MA1, MA3	Opt.
Digital Humanities	MA1, MA3	Opt.
SC master EPFL	MA1, MA3	Opt.

Language	English
Credits	4
Session	Winter
Semester	Fall
Exam	Oral
Workload	120h
Weeks	14
Hours	**4 weekly**
Lecture	2 weekly
Project	2 weekly

Summary

This course addresses the relationship between specific technological features and the learners' cognitive processes. It also covers the methods and results of empirical studies on this topic: do student actually learn due to technologies?

Content

Learning theories and learning processes. Instructional design: methods, patterns and principles. Orchestration graphs. On-line education. Effectiveness of learning technologies. Methods for empirical research. Learning analytics. History of learning technologies.

Keywords

learning, pedagogy, teaching, online education, MOOCs

Learning Prerequisites

Recommended courses

One of these courses is recommended:
- Machine Learning (Jaggi / Urbanke)
- Applied Data Analysis (West)

Learning Outcomes

By the end of the course, the student must be able to:
- Describe the learning processes triggered by a technology-based activity
- Explain how a technology feature influences learning processes
- Elaborate a study that measures the learning effects of a digital environment
- Select appropriately a learning technology given the target audience and the expected learning outcomes
- Apply machine learning methods to educational traces

Transversal skills

- Set objectives and design an action plan to reach those objectives.

Teaching methods

The course will combine participatory lectures with a project around learning analytics

Expected student activities

The project will include several milestones to be delivered along the semester.

Assessment methods

- Project + exam
- 50 / 50

Supervision

Office hours	No
Assistants	Yes
Forum	Yes

Resources

Moodle Link

- http://moodle.epfl.ch/course/view.php?id=14248

CS-423

Distributed information systems

Aberer Karl

Cursus	Sem.	Type
Biocomputing minor	E	Obl.
Communication systems minor	E	Opt.
Computer science minor	E	Opt.
Computer science	MA2	Opt.
Data Science	MA2, MA4	Opt.
Digital Humanities	MA2	Opt.
Electrical and Electronical Engineering	MA2, MA4	Opt.
Energy Management and Sustainability	MA2, MA4	Opt.
Environmental Sciences and Engineering	MA2, MA4	Opt.
SC master EPFL	MA2, MA4	Obl.

Language	English
Credits	4
Session	Summer
Semester	Spring
Exam	Written
Workload	120h
Weeks	14
Hours	**3 weekly**
Lecture	2 weekly
Exercises	1 weekly

Summary

This course introduces the key concepts and algorithms from the areas of information retrieval, data mining and knowledge bases, which constitute the foundations of today's Web-based distributed information systems.

Content

Information Retrieval
1.Information Retrieval - Introduction 2.Text-Based Information Retrieval 3.Vector Space Retrieval 4.Probabilistic Information Retrieval 5.Query Expansion 6.Inverted Index 7.Distributed Retrieval 8.Latent Semantic Indexing 9.Word Embeddings 10. Link-Based Ranking
Data Mining
1.Data Mining – Introduction 2. Association Rule Mining 3. Clustering 4. Classification 5. Mining Social Graphs 6. Classification Methodology 7. Document Classification 8. Recommender Systems
Knowledge Bases
1. Semi-structured data 2. Semantic Web 3. RDF Resource Description Framework 4. Semantic Web Resources 5. Information Extraction 6. Taxonomy Induction 7. Ontology Mapping

Learning Prerequisites

Recommended courses

Introduction to Database Systems

Learning Outcomes

By the end of the course, the student must be able to:

- Characterize the main tasks performed by information systems, namely data, information and knowledge management
- Apply semi-structured data models, their representation through Web standards and algorithms for storing and processing semi-structured data
- Apply fundamental models and techniques of text retrieval and their use in Web search engines
- Apply main categories of data mining techniques, local rules, predictive and descriptive models, and master representative algorithms for each of the categories
- Apply collaborative information management models, like crowd-sourcing, recommender systems, social networks

Teaching methods

Ex cathedra + programming exercises (Python)

Assessment methods

25% Continuous evaluations with bonus system during the semester
75% Final written exam (180 min) during exam session

ENG-466

Distributed intelligent systems

Martinoli Alcherio

Cursus	Sem.	Type
Biocomputing minor	H	Opt.
Computational science and Engineering	MA1, MA3	Opt.
Computer science	MA1, MA3	Opt.
Data Science	MA1, MA3	Opt.
Electrical and Electronical Engineering	MA1, MA3	Opt.
Energy Management and Sustainability	MA1, MA3	Opt.
Environmental Sciences and Engineering	MA1, MA3	Opt.
Microtechnics	MA1, MA3	Opt.
Robotics	MA1	Opt.
SC master EPFL	MA1, MA3	Opt.

Language	English
Credits	5
Withdrawal	Unauthorized
Session	Winter
Semester	Fall
Exam	Written
Workload	150h
Weeks	14
Hours	**5 weekly**
Lecture	2 weekly
Exercises	3 weekly

Summary

The goal of this course is to provide methods and tools for modeling distributed intelligent systems as well as designing and optimizing coordination strategies. The course is a well-balanced mixture of theory and practical activities using simulation and real hardware platforms.

Content

- Introduction to key concepts such as self-organization and software and hardware tools used in the course
- Examples of natural, artificial and hybrid distributed intelligent systems
- Modeling methods: sub-microscopic, microscopic, macroscopic, multi-level; spatial and non-spatial; mean field, approximated and exact approaches
- Machine-learning methods: single- and multi-agent techniques; expensive optimization problems and noise resistance
- Coordination strategies and distributed control: direct and indirect schemes; algorithms and methods; performance evaluation
- Application examples in distributed sensing and action

Keywords

Artificial intelligence, swarm intelligence, distributed robotics, sensor networks, modeling, machine-learning, control

Learning Prerequisites

Required courses

Fundamentals in analysis, probability, and programming for both compiled and interpreted languages

Recommended courses

Basic knowledge in statistics, specific programming language used in the course (C and Matlab), and signals and systems

Learning Outcomes

By the end of the course, the student must be able to:

- Design a reactive control algorithm
- Formulate a model at different level of abstraction for a distributed intelligent system
- Analyze a model of a distributed intelligent system
- Analyze a distributed coordination strategy/algorithm
- Design a distributed coordination strategy/algorithm
- Implement code for single robot and multi-robot systems
- Carry out systematic performance evaluation of a distributed intelligent system
- Apply modeling and design methods to specific problems requiring distributed sensing and action
- Optimize a controller or a set of possibly coordinated controllers using model-based or data-driven methods

Transversal skills

- Demonstrate a capacity for creativity.
- Access and evaluate appropriate sources of information.
- Collect data.
- Plan and carry out activities in a way which makes optimal use of available time and other resources.
- Make an oral presentation.
- Write a scientific or technical report.
- Evaluate one's own performance in the team, receive and respond appropriately to feedback.

Teaching methods

Ex-cathedra lectures, assisted exercises with mid-term verification, and a course project involving teamwork

Expected student activities

Attending lectures, carrying out exercises and the course project, and reading handouts.

Assessment methods

Continuous control (50%) with final written exam (50%).

Supervision

Office hours	Yes
Assistants	Yes
Forum	No

Resources

Bibliography

Lecture notes, selected papers and book chapters distributed at each lecture.

Websites

- http://disal.epfl.ch/teaching/distributed_intelligent_systems/

Moodle Link

- https://moodle.epfl.ch/course/view.php?id=15472

Prerequisite for

R&D activities in engineering

CS-525

Foundations and tools for processing tree structured data

Vanoirbeek Christine

Cursus	Sem.	Type
Computer science	MA1, MA3	Opt.
Data Science	MA1, MA3	Opt.

Language	English
Credits	4
Session	Winter
Semester	Fall
Exam	Written
Workload	120h
Weeks	14
Hours	**4 weekly**
Lecture	2 weekly
Project	2 weekly

Summary

The course is about the foundations and tools for processing tree structured data, a prevalent model for representing semi-structured data (SSD) over distributed information networks. It aims at presenting approaches, programming languages and tools for modeling and manipulating tree-structured info

Content

The theoretical part introduces underlying concepts sustaining the approach.
The practical part illustrates the application of the concepts in a concrete context: the development of Web applications that make use of an XML native database (one category of the NoSQL databases) and associated XML languages.
Theoretical foundations
• Tree grammars
• Finite tree automata
Type systems to describe and validate the structure of SSD
• Document Type Definition
• XML Schema
• RELAX NG and Schematron
Querying tree structured data and programming
• Navigation and extraction of information from tree structured data (XPath expressions)
• Tree data transformation (XSLT)
• Query and programmig language (XQuery) incl. Static Type Checking
Application scenario
• Use of a development framework in which all these languages fit

Keywords

Tree-shaped data representation and processing, Foundation of XML types, Tree grammars, XML core technologies, Web applications

Learning Outcomes

By the end of the course, the student must be able to:

• Explain and understand the differences - strenghts and weaknesses - of a tree structured model in comparison with other data models.

• Understand the fundamental principles of a strongly typed language to manipulate tree structured data.

• Use core languages for modeling, querying, repurposing and processing tree structured data.

• Identify situations where information management requirements can be more appropriately dealt with a tree structured data model approach.

• Get a flavor of research ongoing in the domain.

Teaching methods

Ex cathedra lectures and group mini-projects.

Expected student activities

Attend the lectures
Work on mini-project

Assessment methods

Written exam and mini-project evaluation.

MATH-360 **Graph theory**

Vacat .

Cursus	Sem.	Type
Data Science	MA2, MA4	Opt.
Mathematics	BA6	Opt.

Language	English
Credits	5
Session	Summer
Semester	Spring
Exam	Written
Workload	150h
Weeks	14
Hours	**4 weekly**
Lecture	2 weekly
Exercises	2 weekly

Summary

The course aims to introduce the basic concepts and results of modern Graph Theory with special emphasis on those topics and techniques that have proved to be applicable in theoretical computer science and in practice during the past forty years.

Assessment methods

WRITTEN EXAM
Dans le cas de l’art. 3 al. 5 du Règlement de section, l’enseignant décide de la forme de l’examen qu’il communique aux étudiants concernés.

CS-486

Human computer interaction

Pu Faltings Pearl

Cursus	Sem.	Type
Computer science	MA2	Opt.
Data Science	MA2, MA4	Opt.
Digital Humanities	MA2	Opt.
Mineur STAS Chine	E	Opt.
SC master EPFL	MA2, MA4	Opt.

Language	English
Credits	4
Session	Summer
Semester	Spring
Exam	During the semester
Workload	120h
Weeks	14
Hours	**4 weekly**
Lecture	2 weekly
Exercises	1 weekly
Project	1 weekly

Summary

This course starts with a simple premise: if a piece of software is useful, joyful and easy to use, people will want it. We thus teach methods for engaging user experience design.

Content

Basic concepts of human-computer interaction
Introduction to HCI: its aims and goals
Design thinking
Qualitative research
User modeling: persona and empathy diagram
Task analysis
Visual design
Basic concepts of cognitive science
How people reason and mental models
How people learn to use software products
How people perceive the world
How people process information
Prototyping methods for HCI design
Storyboarding
Wireframe prototyping
Interactive prototyping
Video prototyping
Evaluation techniques
Cognitive walkthrough
Heuristic evaluation

Keywords

User experience design, design thinking, usability, design for engaging users, rapid prototyping techniques, evaluation with users, design challenge

Learning Prerequisites

Required courses

Introduction to Visual Computing

Recommended courses

Open to students enrolled in the Master and PhD programs in IC.

Important concepts to start the course

Design software for joyful user experience

Learning Outcomes

- Interview users and elicit their needs using the goal-directed design method
- Design interfaces and intearctions
- Project management: set objectives and device a plan to achieve them
- Group work skills: discuss and identify roles, and assume those roles including leadership
- Communication: writing and argumentation skills
- Design and implement interfaces and intearctions

Teaching methods

Lectures, hands-on practice, design review

Expected student activities

Reading, case studies, peer discussions

Assessment methods

Individual project, group project, presentation

Supervision

Office hours	Yes
Forum	Yes

Resources

Virtual desktop infrastructure (VDI)

No

Bibliography

About Face 3: The Essentials of Interaction Design by Alan Cooper et al. (available as e-book at NEBIS)
100 Things Every Designer Needs to Know about People by Susan Weinschenk (avaiable as e-book at NEBiS)

Ressources en bibliothèque

- About Face 3 / Cooper
- 100 Things Every Designer Needs to Know about People / Weinschenk

EE-451 Image analysis and pattern recognition

Thiran Jean-Philippe

Cursus	Sem.	Type
Bioingénierie	MA2, MA4	Opt.
Data Science	MA2, MA4	Opt.
Génie électrique et électronique	MA2, MA4	Opt.
Robotique	MA2	Opt.
Sciences et technologies du vivant	MA2	Opt.

Language	English
Credits	4
Session	Summer
Semester	Spring
Exam	During the semester
Workload	120h
Weeks	14
Hours	**4 weekly**
Lecture	2 weekly
Practical work	2 weekly

Summary

This course gives an introduction to the main methods of image analysis and pattern recognition.

Content

Introduction
Digital image acquisition and properties.
Pre-processing: geometric transforms, linear filtering, image restoration.
Introduction to Mathematical Morphology
Examples and applications

Segmentation and object extraction
Thresholding, edge detection, region detection.
Segmentation by active contours. Applications in medical image segmentation.

Shape representation and description
Contour-based representation, region-based representation. Morphological skeletons

Shape recognition
Statistical shape recognition, Bayesian classification, linear and non-linear classifiers, perceptrons, neural networks and unsupervised classifiers.
Applications.

Practical works on computers

Learning Prerequisites

Recommended courses

Introduction to signal processing, Image processing

Learning Outcomes

By the end of the course, the student must be able to:

- Use Image pre-processing methods
- Use image segmentation methods
- Choose shape description methods appropriate to a problem
- Use classification methods appropriate to a problem

Transversal skills

- Use a work methodology appropriate to the task.
- Assess one's own level of skill acquisition, and plan their on-going learning goals.
- Make an oral presentation.
- Summarize an article or a technical report.
- Identify the different roles that are involved in well-functioning teams and assume different roles, including leadership roles.

Teaching methods

Ex cathedra and practical work and oral presentation by the students

Assessment methods

Continuous control

Resources

Ressources en bibliothèque

- Reconnaissance des formes et analyse de scènes / Kunt
- Image Processing, Analysis and Machine Vision / Sonka

Prerequisite for

Semester project, Master project, doctoral thesis

COM-402

Information security and privacy

González Troncoso Carmela, Hubaux Jean-Pierre, Oechslin Philippe

Cursus	Sem.	Type
Computational science and Engineering	MA2, MA4	Opt.
Computer and Communication Sciences		Obl.
Computer science	MA2	Obl.
Data Science	MA2, MA4	Obl.
Financial engineering	MA2, MA4	Opt.
SC master EPFL	MA2, MA4	Obl.

Language	English
Credits	6
Session	Summer
Semester	Spring
Exam	Written
Workload	180h
Weeks	14
Hours	**6 weekly**
Lecture	4 weekly
Project	2 weekly

Summary

This course will provide a broad overview of information security and privacy topics, with the primary goal of giving students the knowledge and tools they will need "in the field" in order to deal with the security/privacy challenges they are likely to encounter in today's "Big Data" world.

Content

• Data protection concepts: access control, encryption, compartmentalization
• Intrusion/hacking techniques, intrusion detection, advanced persistent threats
• Practices for management of personally identifying information
• Operational security practices and failures
• Data anonymization and de-anonymization techniques
• Information flow control
• Differential privacy
• Cryptographic tools for data security and privacy
• Policy, ethics, and legal considerations

Keywords

security, privacy, protection, intrusion, anonymization, cryptography

Learning Prerequisites

Required courses

Basic programming course or comparable demonstration of basic programming skills

Learning Outcomes

By the end of the course, the student must be able to:

- Understand the most important classes of information security/privacy risks in today's â##Big Dataâ## environment
- Exercise a basic, critical set of â##best practicesâ## for handling sensitive information
- Exercise competent operational security practices in their home and professional lives
- Understand at overview level the key technical tools available for security/privacy protection

Expected student activities

Attending lectures, solving assigned problems and "hands-on" exercises, reading and demonstrating understanding of provided materials.

Assessment methods

Continuous assessment via homework exercises, and final written exam.

COM-404

Information theory and coding

Telatar Emre

Cursus	Sem.	Type
Communication systems minor	H	Opt.
Computer and Communication Sciences		Obl.
Computer science minor	H	Opt.
Computer science	MA1, MA3	Opt.
Data Science	MA1, MA3	Opt.
Data science minor	E	Opt.
Electrical and Electronical Engineering	MA1, MA3	Opt.
SC master EPFL	MA1, MA3	Obl.

Language	English
Credits	7
Session	Winter
Semester	Fall
Exam	Written
Workload	210h
Weeks	14
Hours	**6 weekly**
Lecture	4 weekly
Exercises	2 weekly

Summary

The mathematical principles of communication that govern the compression and transmission of data and the design of efficient methods of doing so.

Content

1. Mathematical definition of information and the study of its properties.
2. Source coding: efficient representation of message sources.
3. Communication channels and their capacity.
4. Coding for reliable communication over noisy channels.
5. Multi-user communications: multi access and broadcast channels.
6. Lossy source coding : approximate representation of message sources.
7. Information Theory and statistics

Learning Outcomes

By the end of the course, the student must be able to:

- Formulate the fundamenal concepts of information theory such as entropy, mutual information, channel capacity
- Elaborate the principles of source coding and data transmission
- Analyze source codes and channel codes
- Apply information theoretic methods to novel settings

Teaching methods

Ex cathedra + exercises

Assessment methods

With continuous control

Resources

Ressources en bibliothèque

- Elements of Information Theory / Cover

Websites

- http://moodle.epfl.ch/enrol/index.php?id=14593

COM-406

Information theory and signal processing

Gastpar Michael Christoph, Telatar Emre, Urbanke Rüdiger

Cursus	Sem.	Type
Computational science and Engineering	MA1, MA3	Opt.
Data Science	MA1, MA3	Obl.
Data science minor	H	Opt.
Digital Humanities	MA1, MA3	Opt.

Language	English
Credits	6
Session	Winter
Semester	Fall
Exam	Written
Workload	180h
Weeks	14
Hours	**6 weekly**
Lecture	4 weekly
Exercises	2 weekly

Summary

Information Theory and Signal Processing are key underpinnings of Data Science. They provide frameworks for signal representation and for fundamental performance bounds.

Content

This class presents basic concepts of Information Theory and Signal Processing and their relevance to emerging problems in Data Science and Machine Learning.
A tentative list of topics covered is:

1. Signal Representations
2. Measures of Information
3. Compression and Quantization
4. Sparsity
5. Exponential Families, Maximum Entropy
6. Detection and Estimation Theory

Keywords

Information Theory, Signal Processing, Statistical Signal Processing, Machine Learning, Data Science.

Learning Prerequisites

Required courses

COM-300 Modèles stochastiques pour les communications

Recommended courses

Statistics

Important concepts to start the course

Solid understanding of linear algebra and probability as well as real and complex analysis.

Learning Outcomes

By the end of the course, the student must be able to:

- Formulate the fundamental concepts of signal processing such as basis representations and sampling
- Formulate the fundamental concepts of information theory such as entropy and mutual information
- Analyze problems in statistical settings using fundamental bounds from information theory
- Formulate problems using robust and universal techniques

Teaching methods

Ex cathedra lectures, exercises, and small projects.

Expected student activities

Follow lectures; independent work on problems (homework and small projects).

Assessment methods

Written final exam during the exam session.
Homework Problem Sets during the semester.
10% homework, 90% final exam.

Supervision

Assistants	Yes

Resources

Bibliography

Cover and Thomas, Elements of Information Theory (Second Edition), Wiley, 2006.

Ressources en bibliothèque

- Elements of Information Theory / Cover

Notes/Handbook

Lectures notes

Websites

- https://ipg.epfl.ch/cms/lang/en/pid/147664

CS-430 **Intelligent agents**

Faltings Boi

Cursus	Sem.	Type
Computer and Communication Sciences		Obl.
Computer science minor	H	Opt.
Computer science	MA1, MA3	Opt.
Data Science	MA1, MA3	Opt.
Data science minor	H	Opt.
Energy Management and Sustainability	MA1, MA3	Opt.
Financial engineering	MA1, MA3	Opt.
Robotics	MA1	Opt.
SC master EPFL	MA1, MA3	Opt.

Language	English
Credits	6
Session	Winter
Semester	Fall
Exam	During the semester
Workload	180h
Weeks	14
Hours	**6 weekly**
Lecture	3 weekly
Exercises	3 weekly

Summary

Software agents are widely used to control physical, economic and financial processes. The course presents practical methods for implementing software agents and multi-agent systems, supported by programming exercises, and the theoretical underpinnings including computational game theory.

Content

The course contains 4 main subject areas:

1) Basic models and algorithms for individual agents:
game-playing algorithms, reactive agents and reinforcement learning. Models and algorithms for rational, goal-oriented behavior in agents.
2) Multi-agent systems:
multi-agent planning, distributed algorithms for constraint satisfaction, coordination techniques for multi-agent systems.
3) Self-interested agents:
Models and algorithms for implementing self-interested agents motivated by economic principles: elements of computational game theory, models and algorithms for automated negotiation, social choice, mechanism design, electronic auctions and marketplaces.
4) Implementing multi-agent systems:
Agent platforms, ontologies and markup languages, web services and standards for their definition and indexing.

Learning Prerequisites

Recommended courses

Intelligence Artificielle or another introductory course to AI

Learning Outcomes

By the end of the course, the student must be able to:

- Choose and implement methods for rational decision making in software agents, based on decision processes and AI planning techniques
- Choose and implement methods for efficient rational decision making in teams of multiple software agents
- Model scenarios with multiple self-interested agents in the language of game theory
- Evaluate the feasibility of achieving goals with self-interested agents using game theory
- Design, choose and implement mechanisms for self-interested agents using game theory
- Implement systems of software agents using agent platforms

Teaching methods

Ex cathedra, practical programming exercises

Expected student activities

Lectures: 3 hours
Reading: 3 hours
Assignments/programming: 4 hours

Assessment methods

Mini-projects and exercises 40%, final exam 60%

Resources

Bibliography

Michael Wooldridge : An Introduction to MultiAgent Systems - Second Edition, John Wiley & Sons, 2009Stuart Russell and Peter Norvig: Artificial Intelligence: A Modern Approach (2nd/3rd Edition), Prentice Hall Series in Artificial Intelligence, 2003/2009.

Ressources en bibliothèque

- An Introduction to MultiAgent Systems / Wooldridge
- Artificial Intelligence: A Modern Approach / Russell

Websites

- http://liawww.epfl.ch/
- http://moodle.epfl.ch/

CS-431 **Introduction to natural language processing**

Chappelier Jean-Cédric, Rajman Martin

Cursus	Sem.	Type
Computer science	MA1, MA3	Opt.
Data Science	MA1, MA3	Opt.
Digital Humanities	MA1, MA3	Opt.
SC master EPFL	MA1, MA3	Opt.

Language	English
Credits	4
Session	Winter
Semester	Fall
Exam	Written
Workload	120h
Weeks	14
Hours	**4 weekly**
Lecture	2 weekly
Exercises	2 weekly

Summary

The objective of this course is to present the main models, formalisms and algorithms necessary for the development of applications in the field of natural language information processing. The concepts introduced during the lectures will be applied during practical sessions.

Content

Several models and algorithms for automated textual data processing will be described: (1) morpho-lexical level: electronic lexica, spelling checkers, ...; (2) syntactic level: regular, context-free, stochastic grammars, parsing algorithms, ...; (3) semantic level: models and formalisms for the representation of meaning, ...
Several application domains will be presented: Linguistic engineering, Information Retrieval, Text mining (automated knowledge extraction), Textual Data Analysis (automated document classification, visualization of textual data).

Keywords

Natural Language Processing; Computationnal Linguisitics; Part-of-Speech tagging; Parsing

Learning Outcomes

By the end of the course, the student must be able to:

- Compose key NLP elements to develop higher level processing chains
- Assess / Evaluate NLP based systems
- Choose appropriate solutions for solving typical NLP subproblems (tokenizing, tagging, parsing)
- Describe the typical problems and processing layers in NLP
- Analyze NLP problems to decompose them in adequate independant components

Teaching methods

Ex cathedra ; practical work on computer

Expected student activities

attend lectures and practical sessions, answer quizzes.

Assessment methods

4 quiz during semester 25%, final exam 75%

Resources

Virtual desktop infrastructure (VDI)

No

Bibliography

1. M. Rajman editor, "*Speech and Language Engineering*", EPFL Press, 2006.
2. Daniel Jurafsky and James H, Martin, "*Speech and Language Processing"*, Prentice Hall, 2008 (2nd edition)
3. Christopher D. Manning and Hinrich Schütze, "*Foundations of Statistical Natural Language Processing*", MIT Press, 2000
4. Christopher D. Manning, Prabhakar Raghavan and Hinrich Schütze, *Introduction to Information Retrieval*, Cambridge University Press. 2008
5. Nitin Indurkhya and Fred J. Damerau editors, "*Handbook of Natural Language Processing*", CRC Press, 2010 (2nd edition)

Ressources en bibliothèque

- Handbook of Natural Language Processing / Indurkhya
- Introduction to Information Retrieval / Manning
- Foundations of Statistical Natural Language Processing / Manning
- Speech and Language Engineering / Rajman
- Speech and Language Processing / Jurafsky

Websites

- http://coling.epfl.ch

EE-490(h)

Lab in data science

Verscheure Olivier

Cursus	Sem.	Type
Data Science	MA2, MA4	Opt.
Electrical and Electronical Engineering	MA2, MA4	Opt.

Language	English
Credits	4
Withdrawal	Unauthorized
Session	Summer
Semester	Spring
Exam	During the semester
Workload	120h
Weeks	14
Hours	**4 weekly**
Practical work	4 weekly

Summary

This hands-on course teaches the tools & methods used by data scientists, from researching solutions to scaling up prototypes to Spark clusters. It exposes the students to the entire data science pipeline, from data acquisition to extracting valuable insights applied to real-world problems.

Content

1. Crash-course in Python for data scientists

- Python packages: NumPy, Pandas, Matplotlib, Scikit-Learn
- Interactive data science with web-based notebooks
- **Project #1**: Curating data from a network of CO2 sensors

2. Distributed computing with an Apache Hadoop distribution

- Understand main constituents: HDFS, Parquet, HBase, Hive, Zookeeper, Ambari, Spark, Spark Streaming, Yarn, Mesos, etc.
- **Project #2.1**: Prepare a sandbox distribution

- HDFS internals, best practices
- **Project #2.2**: Configure HDFS, prepare files used in subsequent projects, choose appropriate compression, etc.

3. Distributed processing with Apache Spark

- RDDs and best practices for order of operations, data partitioning, caching
- Data science packages in Spark: GraphX, MLlib, etc.
- **Project #3**: Large-scale processing of genomic data

4. Real-time data acquisition using Apache NiFi

- Stream processing using Apache Spark Streaming
- **Project #4**: Indexing tweets with NiFi and Solr

5. Final project - Summing it all up

- Tapping into live traffic data sources from a major city: Acquisition & curation of live traffic sensors, estimation of speed of traffic on different road segments, and prediction of congestion using Spark, HBase, Kafka.

Keywords

Data Science, IoT, Machine Learning, Predictive Modeling, Big Data, Stream Processing, Apache Spark, Hadoop, Large-Scale Data Analysis

Learning Prerequisites

Required courses

Students must have prior experience with at least one general-purpose programming language.

Important concepts to start the course

It is recommended that students familiarize themselves with concepts in statistics and standard methods in machine learning.

Learning Outcomes

By the end of the course, the student must be able to:

- Use standard Big Data tools and Data Science librairies
- Carry out real-world projects with a variety of real datasets, both at rest and in motion
- Design large scale data science and engineering problems
- Present tangible solution to a real-world Data Science problem

Transversal skills

- Demonstrate a capacity for creativity.
- Plan and carry out activities in a way which makes optimal use of available time and other resources.
- Write a scientific or technical report.

Teaching methods

- Hands-on lab sessions

- Homework assignments
- Final project

... using real-world datasets and Cloud Compute & Storage Services

Expected student activities

Students are expected to:

- STUDY: Attend the lab sessions
- WORK: Complete homework assignments
- ENGAGE: Contribute to the ineractive nature of the class
- COLLABORATE: Work in small groups to provide solutions to real-world problems
- EXPLAIN: Present ideas and results to the class

Assessment methods

- 60% continuous assessment during the semester
- 40% final project, done in small groups

Supervision

Office hours	Yes
Assistants	Yes
Forum	Yes

Resources

Virtual desktop infrastructure (VDI)

No

Bibliography

- **Python Data Science Handbook: Essential Tools for Working with Data** by Jake VanderPlas, O'Reilly Media, November 2016

- pyGAM - https://github.com/dswah/pyGAM

A list of additional readings will be distributed at the beginning of the course.

Ressources en bibliothèque

- Python Data Science Handbook: Essential Tools for Working with Data / J. VanderPlas

Websites

- http://www.datascience.ch

CS-526 **Learning theory**

Macris Nicolas, Svensson Ola Nils Anders, Urbanke Rüdiger

Cursus	Sem.	Type
Computer science	MA2	Opt.
Data Science	MA2, MA4	Opt.
SC master EPFL	MA2, MA4	Opt.

Language	English
Credits	4
Session	Summer
Semester	Spring
Exam	Written
Workload	120h
Weeks	14
Hours	**4 weekly**
Lecture	2 weekly
Exercises	2 weekly

Summary

Machine learning and data analysis are becoming increasingly central in many sciences and applications. This course concentrates on the theoretical underpinnings of machine learning.

Content

- Basics : statistical learning framework, Probably Approximately Correct (PAC) learning, learning with a finite number of classes, Vapnik-Chervonenkis (VC) dimension, non-uniform learnability, complexity of learing.
- Neural Nets : representation power of neural nets, learning and stability, PAC Bayes bounds.
- Graphical model learning.
- Non-negative matrix factorization, Tensor decompositions and factorization.
- Learning mixture models.

Learning Prerequisites

Recommended courses

- Analysis I, II, III
- Linear Algebra
- Machine learning
- Probability
- Algorithms (CS-250)

Learning Outcomes

By the end of the course, the student must be able to:

- Explain the framework of PAC learning
- Explain the importance basic concepts such as VC dimension and non-uniform learnability
- Describe basic facts about representation of functions by neural networks
- Describe recent results on specific topics e.g., graphical model learning, matrix and tensor factorization, learning mixture models

Teaching methods

- Lectures

- Exercises

Expected student activities

- Attend lectures
- Attend exercises sessions and do the homework

Assessment methods

Final exam and graded homeworks

Supervision

Office hours	Yes
Assistants	Yes
Forum	Yes
Others	Course website

MATH-341

Linear models

Thibaud Emeric Rolland Georges

Cursus	Sem.	Type
Data Science	MA1, MA3	Opt.
Digital Humanities	MA1, MA3	Opt.
Mathematics	BA5	Opt.

Language	English
Credits	5
Session	Winter
Semester	Fall
Exam	Written
Workload	150h
Weeks	14
Hours	**4 weekly**
Lecture	2 weekly
Exercises	2 weekly

Summary

Regression modelling is a basic tool of statistics, because it describes how one variable may depend on another. The aim of this course is to familiarize students with the basis of regression modelling, and of some related topics.

Content

- Properties of the multivariate Gaussian distribution and related quadratic forms.
- Gaussian linear regression: likelihood, least squares, geometrical interpretation.
- Distribution theory, confidence and prediction intervals.
- Gauss-Markov theorem.
- Model checking and validation: residual diagnostics, outliers and leverage points.
- Analysis of variance.
- Model selection: bias/variance tradeoff, stepwise procedures, information-based criteria.
- Multicollinearity and penalised estimation: ridge regression, LASSO.
- Robust regression and M-estimation.
- Other topics as time permits: logistic and Poisson regression, nonparametric regression.

Learning Prerequisites

Recommended courses

Analysis, Linear Algebra, Probability, Statistics

Learning Outcomes

By the end of the course, the student must be able to:

- Recognize when a linear model is appropriate to model dependence
- Interpret model parameters both geometrically and in applied contexts
- Estimate the parameters determining a linear model from empirical observations
- Test hypotheses related to the structural characteristics of a linear model
- Construct confidence bounds for model parameters and model predictions
- Analyze variation into model components and error components
- Contrast competing linear models in terms of fit and parsimony
- Construct linear models to balance bias, variance and interpretability
- Assess / Evaluate the fit of a linear model to data and the validity of its assumptions.
- Prove basic results related to the statistical theory of linear models

Teaching methods

Lectures ex cathedra, exercises in class, take-home projects

Assessment methods

Continuous control, final exam.
Seconde tentative : Dans le cas de l'art. 3 al. 5 du Règlement de section, l'enseignant décide de la forme de l'examen qu'il communique aux étudiants concernés.

Supervision

Office hours	No
Assistants	Yes
Forum	Yes

Resources

Virtual desktop infrastructure (VDI)

No

Ressources en bibliothèque

- Statistical Models / Davison
- Methods and Applications of Linear Models / Hocking
- Applied Regression Analysis / Draper

CS-433

Machine learning

Jaggi Martin, Urbanke Rüdiger

Cursus	Sem.	Type
Biocomputing minor	H	Obl.
Communication systems minor	H	Obl.
Computational Neurosciences minor	H	Opt.
Computational science and Engineering	MA1, MA3	Opt.
Computer and Communication Sciences		Obl.
Computer science minor	H	Opt.
Computer science	MA1, MA3	Obl.
Data Science	MA1, MA3	Obl.
Data science minor	H	Opt.
Digital Humanities	MA1, MA3	Opt.
Electrical Engineering		Obl.
Electrical and Electronical Engineering	MA1, MA3	Opt.
Financial engineering	MA1, MA3	Opt.
Life sciences and technologies	MA1	Opt.
Managmt, tech et entr.	MA1, MA3	Opt.
SC master EPFL	MA1, MA3	Obl.
Sciences du vivant	MA1, MA3	Opt.

Language	English
Credits	7
Session	Winter
Semester	Fall
Exam	Written
Workload	210h
Weeks	14
Hours	**6 weekly**
Lecture	4 weekly
Exercises	2 weekly

Summary

Machine learning and data analysis are becoming increasingly central in many sciences and applications. In this course, fundamental principles and methods of machine learning will be introduced, analyzed and practically implemented.

Content

1. *Basic regression and classification concepts and methods: Linear models, overfitting, linear regression, Ridge regression, logistic regression, and k-NN.*
2. *Fundamental concepts: cost-functions and optimization, cross-validation and bias-variance trade-off, curse of dimensionality.*
3. *Unsupervised learning: k-Means Clustering, Gaussian mixture models and the EM algorithm.*
4. *Dimensionality reduction: PCA and matrix factorization, word embeddings*
5. *Advanced methods: generalized linear models, SVMs and Kernel methods, Neural networks and deep learning*

Keywords

- *Machine learning, pattern recognition, deep learning, data mining, knowledge discovery, algorithms*

Learning Prerequisites

Required courses

- Analysis I, II, III
- Linear Algebra
- Probability and Statistics (MATH-232)
- Algorithms (CS-250)

Recommended courses

- *Introduction to differentiable optimization (MATH-265)*
- *Linear Models (MATH-341)*

Important concepts to start the course

- *Basic probability and statistics (conditional and joint distribution, independence, Bayes rule, random variables, expectation, mean, median, mode, central limit theorem)*
- *Basic linear algebra (matrix/vector multiplications, systems of linear equations, SVD)*
- *Multivariate calculus (derivative w.r.t. vector and matrix variables)*
- *Basic Programming Skills (labs will use Python)*

Learning Outcomes

By the end of the course, the student must be able to:

- Define the following basic machine learning problems: Regression, classification, clustering, dimensionality reduction, time-series
- Explain the main differences between them
- Implement algorithms for these machine learning models
- Optimize the main trade-offs such as overfitting, and computational cost vs accuracy
- Implement machine learning methods to real-world problems, and rigorously evaluate their performance using cross-validation. Experience common pitfalls and how to overcome them
- Explain and understand the fundamental theory presented for ML methods

Teaching methods

- Lectures
- Lab sessions
- Course Projects

Expected student activities

Students are expected to:

- attend lectures
- attend lab sessions and work on the weekly theory and coding exercises
- work on projects using the code developed during labs, in small groups

Assessment methods

- Written final exam
- Continuous control (Course projects)

Supervision

Office hours	Yes
Assistants	Yes
Forum	Yes

Resources

Virtual desktop infrastructure (VDI)

No

Bibliography

• Christopher Bishop, Pattern Recognition and Machine Learning
• Kevin Murphy, Machine Learning: A Probabilistic Perspective
• Shai Shalev-Shwartz, Shai Ben-David, Understanding Machine Learning
• Michael Nielsen, Neural Networks and Deep Learning
• (Jerome Friedman, Robert Tibshirani, Trevor Hastie, The elements of statistical learning : data mining, inference, and prediction)

Ressources en bibliothèque

- The elements of statistical learning : data mining, inference, and prediction / Friedman
- Pattern Recognition and Machine Learning / Bishop
- Understanding Machine Learning / Shalev-Shwartz
- Machine Learning: A Probabilistic Perspective / Murphy
- Neural Networks and Deep Learning / Nielsen

Notes/Handbook

github.com/epfml/ML_course

Websites

- https://mlo.epfl.ch/page-146520.html

COM-516 **Markov chains and algorithmic applications**

Lévêque Olivier, Macris Nicolas

Cursus	Sem.	Type
Computer science	MA1, MA3	Opt.
Data Science	MA1, MA3	Opt.
Electrical Engineering		Obl.
SC master EPFL	MA1, MA3	Opt.

Language	English
Credits	4
Session	Winter
Semester	Fall
Exam	Written
Workload	120h
Weeks	14
Hours	**4 weekly**
Lecture	2 weekly
Exercises	2 weekly

Remarque

The same course was given in Spring 2015-2016 under the name "Random Walks".

Summary

The study of random walks finds many applications in computer science and communications. The goal of the course is to get familiar with the theory of random walks, and to get an overview of some applications of this theory to problems of interest in communications, computer and network science.

Content

Part 1: Markov chains (~6 weeks):
- basic properties: irreducibility, periodicity, recurrence/transience, stationary and limiting distributions,
- ergodic theorem: coupling method
- detailed balance
- convergence rate to the equilibrium, spectral gap, mixing times
- cutoff phenomenon

Part 2: Sampling (~6 weeks)
- classical methods, importance and rejection sampling
- Markov Chain Monte Carlo methods, Metropolis-Hastings algorithm, Glauber dynamics, Gibbs sampling
- applications: function minimization, coloring problem, satisfiability problems, Ising models
- coupling from the past and exact simulation

Keywords

random walks, stationarity, ergodic, convergence, spectral gap, mixing time, sampling, Markov chain Monte Carlo, coupling from the past

Learning Prerequisites

Required courses

Basic probability course
Basic linear algebra and calculus courses

Recommended courses

Stochastic Models for Communications (COM-300)

Important concepts to start the course

Good knowledge of probability and analysis.
Having been exposed to the theory of Markov chains.

Learning Outcomes

By the end of the course, the student must be able to:

- Analyze the behaviour of a random walk
- Assess / Evaluate the performance of an algorithm on a graph
- Implement efficiently various sampling methods

Teaching methods

ex-cathedra course

Expected student activities

active participation to exercise sessions and implementation of a sampling algorithm

Assessment methods

midterm, mini-project, written exam

Resources

Bibliography

Various references will be given to the students during the course, according to the topics discussed in class.

Ressources en bibliothèque

- Probability and random processes / Grimmett

Notes/Handbook

Lecture notes will be provided

Websites

- http://ipgold.epfl.ch/~leveque/Markov_Chains/

Prerequisite for

This course is not so to speak a prerequisite for other courses, but could complement well the course COM-512 on Networks out of control, as well as other courses in statistics.

COM-514

Mathematical foundations of signal processing

Kolundzija Mihailo, Parhizkar Reza, Scholefield Adam James

Cursus	Sem.	Type
Communication systems minor	H	Opt.
Computational science and Engineering	MA1, MA3	Opt.
Computer and Communication Sciences		Obl.
Computer science	MA1, MA3	Opt.
Data Science	MA1, MA3	Opt.
SC master EPFL	MA1, MA3	Opt.
Systems Engineering minor	H	Opt.

Language	English
Credits	6
Session	Winter
Semester	Fall
Exam	Written
Workload	180h
Weeks	14
Hours	**5 weekly**
Lecture	3 weekly
Exercises	2 weekly

Summary

Signal processing tools are presented from an intuitive geometric point of view which is at the heart of all modern signal processing techniques. Student will develop the mathematical depth and rigor needed for the study of advanced topics in signal processing.

Content

From Euclid to Hilbert applied to inverse problems (vector spaces; Hilbert spaces; approximations, projections and decompositions; bases)
Sequences, Discrete-Time Systems, Functions and Continuous-Time Systems (flipped class review of discrete-time Fourier transform; z-transform; DFT; Fourier transform and Fourier series).
Sampling and Interpolation (sampling and interpolation with finite-dimensional vectors, sequences and functions)

Computerized tomography fundamentals (line integrals and projections, Radon transform, Fourier projection/slice theorem, filtered backprojection algorithm, algebraic reconstruction techniques).

Array signal processing fundamentals (spatial filtering and beamforming, adaptive beamforming, acoustic and EM source localization techniques).

Compressed sensing and finite rate of innovation (overview and definitions, reconstruction methods and applications)
Euclidean Distance Matrices (definition, properties and applications).

Learning Prerequisites

Required courses

Circuits and Systems
Signal processing for communications (or Digital signal processing on Coursera)

Learning Outcomes

By the end of the course, the student must be able to:

- Master the right tools to tackle advanced signal and data processing problems
- Develop an intuitive understanding of signal processing through a geometrical approach
- Get to know the applications that are of interest today
- Learn about topics that are at the forefront of signal processing research

Teaching methods

Ex cathedra with exercises
One week of flipped class

Expected student activities

Attending lectures, completing exercises

Assessment methods

Homeworks 20%, midterm (written) 30%, final exam (written) 50%

Supervision

Office hours	Yes
Assistants	Yes
Forum	No

Resources

Virtual desktop infrastructure (VDI)

No

Bibliography

M. Vetterli, J. Kovacevic and V. Goyal, *"Signal Processing: Foundations"*, Cambridge U. Press, 2014.
Available in open access at http://www.fourierandwavelets.org

Ressources en bibliothèque

- Signal Processing: Foundations / Vetterli

Websites

- http://lcav.epfl.ch/SP_Foundations

Moodle Link

- http://moodle.epfl.ch/course/view.php?id=13431

EE-556

Mathematics of data: from theory to computation

Cevher Volkan

Cursus	Sem.	Type
Data Science	MA1	Opt.
Génie électrique et électronique	MA1, MA3	Obl.
Génie électrique		Obl.
Managmt, tech et entr.	MA1, MA3	Opt.
Science et ing. computationelles	MA1, MA3	Opt.

Language	English
Credits	4
Session	Winter
Semester	Fall
Exam	Written
Workload	120h
Weeks	14
Hours	**4 weekly**
Lecture	2 weekly
Exercises	2 weekly

Summary

This course reviews recent advances in convex optimization and statistical analysis in the wake of Big Data. We provide an overview of the emerging convex formulations and their guarantees, describe scalable solution techniques, and illustrate the role of parallel and distributed computation.

Content

The course consists of the following topics
Lecture 1: “Objects in Space”: Definitions of norms, inner products, and metrics for vector, matrix and tensor objects. Basics of complexity theory.
Lecture 2: Maximum likelihood principle as a motivation for convex optimization. Fundamental structures in convex analysis, such as cones, smoothness, and conjugation.
Lecture 3: Unconstrained, smooth minimization techniques. Gradient methods. Variable metric algorithms. Time-data tradeoffs in ML estimation.
Lecture 4: Convex geometry of linear inverse problems. Structured data models (e.g., sparse and low-rank) and convex gauge functions and formulations that encourage these structures. Computational aspects of gauge functions.
Lecture 5: Composite convex minimization. Regularized M-estimators. Time-data tradeoffs in linear inverse problems.
Lecture 6: Convex demixing. Statistical dimension. Phase transitions in convex minimization. Smoothing approaches for non-smooth convex minimization.
Lecture 7: Constrained convex minimization-I. Introduction to convex duality. Classical solution methods (the augmented Lagrangian method, alternating minimization algorithm, alternating direction method of multipliers, and the Frank-Wolfe method) and their deficiencies
Lecture 8: Constrained convex minimization-II. Variational gap characterizations and dual smoothing. Scalable, black-box optimization techniques. Time data-tradeoffs for linear inverse problems.
Lecture 9: Classical black-box convex optimization techniques. Linear programming, semidefinite programming, and the interior point method (IPM). Hierarchies of classical formulations. Time and space complexity of the IPM.
Lecture 10: Time-data tradeoffs in machine learning.
Lecture 11: Convex methods for Big Data I: Randomized coordinate descent methods. The Page Rank problem and Nesterov’s solution. Composite formulations.
Lecture 12: Convex methods for Big Data II: Stochastic gradient descent methods. Least squares: conjugate gradients vs. a simple stochastic gradient method. Dual and gradient averaging schemes. Stochastic mirror descent.
Lecture 13: Randomized linear algebra routines for convex optimization. Probabilistic algorithms for constructing approximate low-rank matrix decompositions. Subset selection approaches. Theoretical approximation guarantees.
Lecture 14: Role of parallel and distributed computing. How to avoid communication bottlenecks and synchronization. Consensus methods. Memory lock-free, decentralized, and asynchronous algorithms.

Learning Prerequisites

Important concepts to start the course

Previous coursework in calculus, linear algebra, and probability is required.
Familiarity with optimization is useful.

Learning Outcomes

By the end of the course, the student must be able to:

- Choose an appropriate convex formulation for a data analytics problem at hand
- Estimate the underlying data size requirements for the correctness of its solution
- Implement an appropriate convex optimization algorithm based on the available computational platform
- Decide on a meaningful level of optimization accuracy for stopping the algorithm
- Characterize the time required for their algorithm to obtain a numerical solution with the chosen accuracy

Assessment methods

Homework assignments. (Continuous control)

COM-512 **Networks out of control**

Cursus	Sem.	Type
Computer science	MA2	Opt.
Data Science	MA2, MA4	Opt.
SC master EPFL	MA2, MA4	Opt.
Systems Engineering minor	E	Opt.

Language	English
Credits	4
Session	Summer
Semester	Spring
Exam	Written
Workload	120h
Weeks	14
Hours	**3 weekly**
Lecture	2 weekly
Exercises	1 weekly

Remarque

Pas donné en 2018-19 - Cours biennal donné les années impaires

Summary

The goal of this class is to acquire mathematical tools and engineering insight about networks whose structure is random, as well as decentralized processes that take place on these networks.

Content

- Course Introduction: Tree Percolation, Branching Processes
- Random Graphs 1: Models, Threshold Functions, Appearance of Subgraphs
- Random Graphs 2: Giant Component and Connectivity
- Random Graphs 3: Other models: the Random Regular Graph, Small World Networks, Scale-Free Networks.
- Random Geometric Graphs: Introduction to Percolation Theory.
- Evolution, Dynamics and Inference 1: Epidemics, Network and Source Discovery.
- Evolution, Dynamics and Inference 2: Information Cascades.
- Evolution, Dynamics and Inference 3: Network Navigation and Price of Anarchy.
- Applications 1: Network Formation Games.
- Applications 2: Homophily, Structural Balance.

Keywords

Random graphs, percolation theory, social networks, communication networks.

Learning Prerequisites

Required courses

Stochastic models in communication (COM-300), or equivalent.

Important concepts to start the course

Basic probability and stastistics; Markov chains; basic combinatorics.

Learning Outcomes

By the end of the course, the student must be able to:

- Analyze social and communication systems

• Model such systems as stochastic models
• Compute key properties of these models

Teaching methods

Ex cathedra lectures, exercises, mini-project

Expected student activities

Attending lectures, bi-weekly homeworks, mini-project incl. student presentation at the end of semester, final exam.

Assessment methods

1. Homeworks 10%
2. Mini-project 40%
3. Final exam 50%.

Supervision

Office hours	Yes
Assistants	Yes
Forum	No

Resources

Bibliography

• A. D. Barbour, L. Holst and S. Janson, Poisson Approximation, Oxford Science Publications, 1992.
• B. Bollobas, Random Graphs (2nd edition), Cambridge University Press, 2001.
• R. Durrett, Random Graph Dynamics, Cambridge University Press, 2006 (electronic version).
• D. Easley, J. Kleinberg. Networks, Crowds, and Markets: Reasoning About a Highly Connected World, Cambridge University Press, 2010 (electronic version).
• G. Grimmett, Percolation (2nd edition), Springer, 1999.
• S. Janson, T. Luczak, A. Rucinski, Random Graphs, Wiley, 2000.
• R. Meester and R. Roy, Continuum Percolation, Cambridge University Press, 1996.

Ressources en bibliothèque

• Random Graphs / Bollobas
• Random Graphs / Janson
• Continuum Percolation / Meester
• Percolation / Grimmett
• Networks, Crowds and Markets / Easley
• Poisson Approximation / Barbour
• Random Graph Dynamics / Durrett

Notes/Handbook

Class notes will be available on the course website.

Websites

• http://icawww1.epfl.ch/class-nooc/

CS-439

Optimization for machine learning

Jaggi Martin

Cursus	Sem.	Type
Computational science and Engineering	MA2, MA4	Opt.
Computer science	MA2	Opt.
Data Science	MA2, MA4	Obl.
Data science minor	E	Opt.
SC master EPFL	MA2, MA4	Opt.

Language	English
Credits	4
Session	Summer
Semester	Spring
Exam	Written
Workload	120h
Weeks	14
Hours	**4 weekly**
Lecture	2 weekly
Exercises	2 weekly

Summary

This course teaches an overview of modern optimization methods, for applications in machine learning and data science. In particular, scalability of algorithms to large datasets will be discussed in theory and in implementation.

Content

This course teaches an overview of modern optimization methods, for applications in machine learning and data science. In particular, scalability of algorithms to large datasets will be discussed in theory and in implementation.
Basic Contents:
Convexity, Gradient Methods, Proximal algorithms, Stochastic and Online Variants of mentioned methods, Coordinate Descent Methods, Subgradient Methods, Frank-Wolfe, Accelerated Methods, Primal-Dual context and certificates, Lagrange and Fenchel Duality, Second-Order Methods, Quasi-Newton Methods. Gradient-Free and Zero-Order Optimization.
Advanced Contents:
Parallel and Distributed Optimization Algorithms, Synchronous and Asynchronous Communication.
Lower Bounds.
Non-Convex Optimization: Convergence to Critical Points, Saddle-Point methods, Alternating minimization for matrix and tensor factorizations
An optional, graded, mini-project allows to explore the real-world performance aspects of the algorithms and variants of the course.

Keywords

Optimization, Machine learning

Learning Prerequisites

Recommended courses

• CS-433 Machine Learning

Important concepts to start the course

• Previous coursework in calculus, linear algebra, and probability is required.
• Familiarity with optimization and/or machine learning is useful.

Learning Outcomes

By the end of the course, the student must be able to:

• Assess / Evaluate the most important algorithms, function classes, and algorithm convergence guarantees

- Compose existing theoretical analysis with new aspects and algorithm variants.
- Formulate scalable and accurate implementations of the most important optimization algorithms for machine learning applications
- Characterize trade-offs between time, data and accuracy, for machine learning methods

Transversal skills

- Use both general and domain specific IT resources and tools
- Summarize an article or a technical report.

Teaching methods

- Lectures
- Exercises with Theory and Implementation Assignments

Expected student activities

Students are expected to:

- Attend the lectures and exercises
- Give a short scientific presentation about a research paper
- Read / watch the pertinent material
- Engage during the class, and discuss with other colleagues

Assessment methods

- Final Exam

Supervision

Office hours	Yes
Assistants	Yes
Forum	Yes

Resources

Virtual desktop infrastructure (VDI)

No

Websites

- https://github.com/epfml/OptML_course

COM-508

Optional project in data science

Profs divers *

Cursus	Sem.	Type
Data Science	MA1, MA2, MA3, MA4	Opt.

Language	English
Credits	8
Session	Winter, Summer
Semester	Fall
Exam	During the semester
Workload	240h
Weeks	14
Hours	**2 weekly**
Project	2 weekly

Summary

Individual research during the semester under the guidance of a professor or an assistant.

Content

Subject to be chosen among the themes proposed on the web site :
http://ic.epfl.ch/data-science-projet-labo-master

Learning Outcomes

By the end of the course, the student must be able to:

- Organize a project
- Assess / Evaluate one's progress through the course of the project
- Present a project

Transversal skills

- Write a literature review which assesses the state of the art.
- Write a scientific or technical report.

Teaching methods

Individual and independant work, under the guidance of a professor or an assistant.

Assessment methods

Oral presentation and written report.

Resources

Websites

- http://ic.epfl.ch/systemes-communication-projet-labo-master_1_1

COM-503 **Performance evaluation**

Le Boudec Jean-Yves

Cursus	Sem.	Type
Computer and Communication Sciences		Opt.
Computer science	MA2	Opt.
Data Science	MA2, MA4	Opt.
Data science minor	E	Opt.
SC master EPFL	MA2, MA4	Opt.

Language	English
Credits	7
Session	Summer
Semester	Spring
Exam	Written
Workload	210h
Weeks	14
Hours	**6 weekly**
Lecture	3 weekly
Exercises	1 weekly
Project	2 weekly

Remarque

cours biennal donné les années paires

Summary

In this course you will learn the methods and techniques that are used to perform a good performance evaluation during a research or development project.

Content

MethodologyA Performance Evaluation Methodology. The scientific method. Dijkstra and Occam's principle.
Statistics and Modeling.
Statistics and modeling, why and how. Comparing systems using sampled data. Regression models. Factorial analysis. Stochastic load and system models. Load forecasting. The Box-Jenkins method.
Practicals.
Using a statistics package (Matlab). Measurements. Discrete event simulation. Stationarity and Steady State. Analysis of simulation results. Perfect Simulations.
Elements of a Theory of Performance. Performance of systems with waiting times. Utilization versus waiting times. Operational laws. Little's formula. Forced flows.law. Stochastic modeling revisited. The importance of the viewpoint. Palm calculus. Application to Simulation Performance patterns in complex systems. Bottlenecks. Congestion phenomenon. Performance paradoxes.

Mini-Project proposed by student.

Learning Prerequisites

Required courses

A first course on probability
A first course on programming

Learning Outcomes

By the end of the course, the student must be able to:

- Estimate confidence intervals
- Design a simulation method
- Critique performance metrics and factors
- Organize a performance evaluation study
- Quantify performance
- Conduct a performance analysis

- Synthesize performance results
- Systematize factors and metrics
- Present results of a performance analysis

Transversal skills

- Use a work methodology appropriate to the task.
- Demonstrate the capacity for critical thinking

Teaching methods

Lectures + pencil and paper exercises + labs + miniproject

Expected student activities

Lectures
Paper and pencil exercises
Labs
Miniproject (last 4 weeks)
Tests every other week

Assessment methods

T = Average of best (n-1) tests done every other week except during miniproject period
E = grade at final exam (during exam session)
L = average of labs
M = miniproject grade

Final grade = 1/4 (T+E+L+M), rounded to the nearest half integer.
All grades except the final grade are not rounded.

Resources

Virtual desktop infrastructure (VDI)

No

Bibliography

- Performance Evaluation of Computer and Communication Systems, Le Boudec Jean-Yves, EPFL Press 2010
- also freely available online at perfeval.epfl.ch

Ressources en bibliothèque

- Performance evaluation of computer and communication systems / Le Boudec

Websites

- http://perfeval.epfl.ch

Moodle Link

- http://moodle.epfl.ch/course/view.php?id=14395

COM-412 **Projet de semestre en data science**

Profs divers *

Cursus	Sem.	Type
Data Science	MA1, MA2, MA3, MA4	Obl.

Langue	français
Crédits	12
Session	Hiver, Eté
Semestre	Automne
Examen	Pendant le semestre
Charge	360h
Semaines	14
Heures	**2 hebdo**
Projet	2 hebdo

Résumé

Travaux de recherche individuelle à effectuer pendant le semestre, selon les directives d'un professeur ou d'un assistant.

Contenu

Sujet de travail à choisir parmil les domaines proposés sur le site web :
http://ic.epfl.ch/data-science-projet-labo-master

Acquis de formation

A la fin de ce cours l'étudiant doit être capable de:

- Organiser un projet
- Evaluer sa progression au cours du projet
- Représenter un projet

Compétences transversales

- Ecrire une revue de la littérature qui établit l'état de l'art.
- Ecrire un rapport scientifique ou technique.

Méthode d'évaluation

Rapport écrit et présentation orale.

Ressources

Sites web

- http://ic.epfl.ch/data-science-projet-semestre

MATH-447

Risk, rare events and extremes

Davison Anthony C.

Cursus	Sem.	Type
Data Science	MA1, MA3	Opt.
Data science minor	H	Opt.
Financial engineering	MA1, MA3	Opt.
Ing.-math	MA1, MA3	Opt.
Mathematics for teaching	MA1, MA3	Opt.
Mathématicien	MA1, MA3	Opt.

Language	English
Credits	5
Session	Winter
Semester	Fall
Exam	Written
Workload	150h
Weeks	14
Hours	**4 weekly**
Lecture	2 weekly
Exercises	2 weekly

Remarque

Cous donné en alternance tous les deux ans (donné en 2018-19)

Summary

Modelling of rare events, such as stock market crashes, storms and catastrophic structural failures, is important. This course will describe the special models and methods that are relevant to such modelling, including the mathematical bases, statistical tools and applications.

Content

- **Mathematical bases**: behaviour of maxima and threshold exceedances in large samples, both for independent and dependent data. Poisson process modelling.

- **Statistical methods**: modelling using the GEV and GP distributions, for independent and dependent data. Likelihood and Bayesian inference. Non-stationarity. Extremal coefficients. Multivariate extreme-value distributions. Max-stable processes.

- **Applications**: Environmental, financial, and engineering applications. Use of R for extremal modelling.

Learning Prerequisites

Important concepts to start the course

Probability and statistics at the level of second-year bachelor (mathematics), plus further knowledge of statistics and stochastic processes.

Learning Outcomes

By the end of the course, the student must be able to:

- Recognize situations where statistical analysis of extrema is appropriate
- Manipulate mathematical objects related to the study of extrema
- Analyze empirical data on extremes using appropriate statistical methods
- Construct appropriate statistical models for extremal data
- Interpret such models in terms of underlying phenomena
- Infer properties of real systems in terms of probability models for extremes

Teaching methods

Lectures, theoretical and computational exercises in class and at home.

Assessment methods

Mini-project, final exam.
Dans le cas de l’art. 3 al. 5 du Règlement de section, l’enseignant décide de la forme de l’examen qu’il communique aux étudiants concernés.

Supervision

Assistants	Yes

Resources

Bibliography

Coles, S. G. (2001) An Introduction to the Statistical Modelling of Extreme Values. Springer.
Beirlant, J, Goegebeur. Y., Teugels. J. and Segers. J. (2004) Statistics of Extremes: Theory and Applications. Wiley.

MATH-441

Robust and nonparametric statistics

Cursus	Sem.	Type
Data Science	MA2, MA4	Opt.
Ing.-math	MA2, MA4	Opt.
Mathématicien	MA2	Opt.

Language	English
Credits	5
Session	Summer
Semester	Spring
Exam	Oral
Workload	150h
Weeks	14
Hours	**4 weekly**
Lecture	2 weekly
Exercises	2 weekly

Remarque

Cours donné en alternance sur deux ans (pas donné en 2018-19)

Summary

In the decades from 1930 to 1950, many rank-based statistics were introduced. These methods were received with much interest, because they worked under weak conditions. Starting in the late 1950, a theory of robustness was added. The course gives an overview of these two theories.

Content

I. Robust Statistics

- Global and local robustness indicators: Breakdown point, influence function
- Hampel's lemma
- Huber's theory: M-estimators, L-estimators
- Robust tests
- Robust regression

II. Linear Rank Tests

- Test of Mann-Whitney-Wilcoxon and general linear rank tests: asymptotic theory, R-estimators
- Rank correlations
- U-statistics
- Comparison of tests: Pitman efficacy
- Permutation tests

III. Estimation of smooth functions

- Curve fitting: polynomial regression, splines
- Smoothing: non parametric estimation, degree of smoothness, bias vs. variance, penalization
- Kernel estimators: definition, properties
- Smoothing splines
- Local regression
- Wavelets

Learning Prerequisites

Required courses

Introduction to Probability, Introduction to Statistics

Learning Outcomes

By the end of the course, the student must be able to:

- Expound the content of the course.
- Apply the statistical methods explained in the course.
- Sketch the proofs of the theoretical results given in the course.
- Choose the appropriate robust or non parametric methods for a given data analysis problem.
- Differentiate between robust and non-parametric methods.
- Generalize the tools treated in the course to other problems.
- Apply spline and kernel smoothers.
- Apply M-estimatiors in a variety of situations.

Transversal skills

- Assess one's own level of skill acquisition, and plan their on-going learning goals.
- Manage priorities.

Teaching methods

Ex cathedra lecture and exercises in the classroom

Expected student activities

Do all the exercices. Prepare each week for the course. Participate actively in the course.

Assessment methods

Oral exam.

Resources

Bibliography

Introduction to the theory of nonparametric statistics by R.H. Randles and D.A. Wolfe, Wiley.
All of nonparametric statistics by L. Wasserman, Springer.
Robust Statistics: The approach based on influence functions by F.R. Hampel, E.M. Ronchetti, P.J. Rousseeuw, W.A. Stahel, Wiley.
Robust Statistics by P.J. Huber, Wiley (second edition).

Ressources en bibliothèque

- Robust Statistics / Huber
- Introduction to the theory of nonparametric statistics / Rhandles
- Robust Statistics / Hampel
- All of nonparametric statistics / Wasserman

MATH-442

Statistical theory

Dehaene Guillaume Philippe Ivan Joseph

Cursus	Sem.	Type
Data Science	MA1, MA3	Opt.
Ing.-math	MA1, MA3	Opt.
Mathematics for teaching	MA1, MA3	Opt.
Mathématicien	MA1, MA3	Opt.

Language	English
Credits	5
Session	Winter
Semester	Fall
Exam	Written
Workload	150h
Weeks	14
Hours	**4 weekly**
Lecture	2 weekly
Exercises	2 weekly

Summary

The course aims to develop certain key aspects of the theory of statistics, providing a common general framework for statistical methodology. While the main emphasis will be on the mathematical aspects of statistics, an effort will be made to balance rigor and relevance to statistical practice.

Content

- Stochastic convergence and its use in statistics: modes of convergence, weak law of large numbers, central limit theorem
- Formalization of a statistical problem : parameters, models, parametrizations, sufficiency, ancillarity, completeness
- Point estimation: methods of estimation, bias, variance, relative efficiency
- Likelihood theory: the likelihood principle, asymptotic properties, misspecification of models, the Bayesian perspective
- Optimality: decision theory, minimum variance unbiased estimation, Cramér-Rao lower bound, efficiency, robustness
- Testing and Confidence Regions: Neyman-Pearson setup, likelihood ratio tests, UMP tests, duality with confidence intervals, confidence regions, large sample theory, goodness-of-fit testing

Learning Prerequisites

Recommended courses

Real Analysis, Linear Algebra, Probability, Statistics

Learning Outcomes

By the end of the course, the student must be able to:

- Formulate the various elements of a statistical problem rigorously.
- Formalize the performance of statistical procedures through probability theory.
- Systematize broad classes of probability models and their structural relation to inference
- Construct efficient statistical procedures for point/interval estimation and testing in classical contexts.
- Derive certain exact (finite sample) properties of fundamental statistical procedures
- Derive Derive certain asymptotic (large sample) properties of fundamental statistical procedures.
- Formulate fundamental limitations and uncertainty principles of statistical theory.
- Prove certain fundamental structural and optimality theorems of statistics.

Teaching methods

Lecture ex cathedra, exercises in class, homework

Assessment methods

Written exam
Dans le cas de l'art. 3 al. 5 du Règlement de section, l'enseignant décide de la forme de l'examen qu'il communique aux étudiants concernés.

Resources

Ressources en bibliothèque

- Mathematical Statistics (e-book)
- Mathematical Statistics / Knight

MATH-413

Statistics for data science

Panaretos Victor

Cursus	Sem.	Type
Computational science and Engineering	MA1, MA3	Opt.
Data Science	MA1, MA3	Obl.
Data science minor	H	Opt.
Electrical Engineering		Obl.
Electrical and Electronical Engineering	MA1, MA3	Opt.
Managmt, tech et entr.	MA1, MA3	Opt.

Language	English
Credits	6
Session	Winter
Semester	Fall
Exam	Written
Workload	180h
Weeks	14
Hours	**6 weekly**
Lecture	4 weekly
Exercises	2 weekly

Summary

Statistics lies at the foundation of data science, providing a unifying theoretical and methodological backbone for the diverse tasks enountered in this emerging field. This course rigorously develops the key notions and methods of statistics, with an emphasis on concepts rather than techniques.

Content

Keywords

Data science, inference, likelihood, regression, regularisation, statistics.

Learning Prerequisites

Required courses

Real analysis, linear algebra, probability.

Recommended courses

A first course in statistics.

Important concepts to start the course

Students taking the course will need a solid grasp of notions from analysis (limits, sequences, series, continuity, differential/integral calculus) and linear algebra (linear subspaces, bases, dimension, eigendecompositions, etc). Though the course will cover a rapid review of probability, a first encounter with the subject is necessary (random variables, distributions/densities, independence, conditional probability). Familiarity with introductory level notions of statistics would be highly beneficial but not necessary.

Learning Outcomes

By the end of the course, the student must be able to:

- Derive properties of fundamental statistical procedures
- Estimate model parameters from empirical observations
- Test hypotheses related to the structural characteristics of a model
- Construct confidence bounds for model parameters and predictions
- Contrast competing models in terms of fit and parsimony

Assessment methods

Final exam.

Resources

Bibliography

Davison, A.C. (2003). Statistical Models, Cambridge.
Panaretos, V.M. (2016). Statistics for Mathematicians. Birkhäuser.
Wasserman, L. (2004). All of Statistics. Springer.
Friedman, J., Hastie, T. and Tibshirani, R. (2010). Elements of Statistical Learning. Springer

COM-506

Student seminar: security protocols and applications

Oechslin Philippe, Vaudenay Serge

Cursus	Sem.	Type
Cyber security minor	E	Opt.
Data Science	MA2, MA4	Opt.
SC master EPFL	MA2, MA4	Opt.

Language	English
Credits	3
Session	Summer
Semester	Spring
Exam	Written
Workload	90h
Weeks	14
Hours	**2 weekly**
Lecture	2 weekly

Summary

This seminar introduces the participants to the current trends, problems, and methods in the area of communication security.

Content

We will look at today's most popular security protocols and new kinds of protocols, techniques, and problems that will play an emerging role in the future. Also, the seminar will cover methods to model and analyze such security protocols. This course will be held as a seminar, in which the students actively participate. The talks will be assigned in the first meeting to teams of students, and each team will have to give a 45 minutes talk, react to other students' questions, and write a 3-4 pages summary of their talk.

Keywords

network security, security protocols, cryptography

Learning Prerequisites

Required courses

- Network security (COM-301)
- Cryptography and security (COM-401)

Learning Outcomes

By the end of the course, the student must be able to:

- Synthesize some existing work on a security protocol
- Analyze a security protocol
- Present a lecture

Transversal skills

- Make an oral presentation.
- Summarize an article or a technical report.

Expected student activities

- prepare a lecture (presentation and a 4-page report)

- present the lecture
- attend to others' lectures and grade them
- do the final exam

Assessment methods

- lecture and attendance to others' lectures (50%)
- final exam (50%)

Supervision

Office hours	No
Assistants	Yes
Forum	No
Others	Lecturers and assistants are available upon appointment.

Resources

Websites

- http://lasec.epfl.ch/teaching.shtml

CS-449 Systems for data science

Koch Christoph

Cursus	Sem.	Type
Computational science and Engineering	MA2, MA4	Opt.
Data Science	MA2, MA4	Obl.
Data science minor	E	Opt.

Language	English
Credits	6
Session	Summer
Semester	Spring
Exam	During the semester
Workload	180h
Weeks	14
Hours	**6 weekly**
Lecture	2 weekly
Exercises	2 weekly
Project	2 weekly

Summary

The course covers fundamental principles for understanding and building systems for managing and analyzing large amounts of data.

Content

Programming methods, including parallel programming:

- *Data-parallel programming: Collection abstractions and modern collection libraries.*

- *Data-flow parallelism vs. message passing. The bulk-synchronous parallel programming model.*
- *SQL and relational algebra. Expressing advanced problems as queries.*

Big data systems design and implementation:

- *Scalability. Synchrony. Distributed systems architectures.*
- *Data locality. Memory hierarchies. New hardware. Sequential versus random access to secondary storage. Partitioning and replication. Data layouts – column stores.*
- *Massively parallel processing operations – joins and sorting*
- *Query optimization. Index selection. Physical database design. Database tuning.*
- *Challenges of big data machine learning systems.*

Changing data:

- *Introduction to transaction processing: purpose, anomalies serializability; concurrency*
- *Commits and consensus.*
- *Eventual consistency. The CAP theorem. NoSQL and NewSQL systems.*

Online / Streaming / Real-time analytics:

- *Data stream processing. Windows. Load shedding.*
- *"Small data"/online aggregation: Sampling and approximating aggregates.*
- *Incremental and online query processing: incremental view maintenance and materialized views.*
-
Data warehousing: The data warehousing workflow, ETL. OLAP, Data Cubes

Keywords

Databases, data-parallel programming, NoSQL systems, query processing.

Learning Prerequisites

Required courses

CS-322: Introduction to database systems

Recommended courses

CS-323: Introduction to operating systems

CS-206 Parallelism and concurrency

Important concepts to start the course

- *Algorithms and data structures – sorting algorithms, balanced trees, graph traversals.*
- *The Scala programming language will be used throughout the course. Programming experience in this language is strongly recommended.*
- *Basic knowledge or computer networking and distributed systems*

Learning Outcomes

By the end of the course, the student must be able to:

- Choose systems parameters, data layouts, query plans, and application designs for database systems and applications.
- Develop data-parallel analytics programs that make use of modern clusters and cloud offerings to scale up to very large workloads.
- Analyze the trade-offs between various approaches to large-scala data management and analytics, depending on efficiency, scalability, and latency needs
- Choose the most appropriate existing systems architecture and technology for a task

Teaching methods

Ex cathedra; including exercises in class, practice with pen and paper or with a computer, and a project

Expected student activities

During the semester, the students are expected to:

- attend the lectures in order to ask questions and interact with the professor,
- attend the exercises session to solve and discuss exercises,
- solve practical homeworks and/or finish a project during the semester,
- take a midterm
- take a final exam

Assessment methods

Homeworks, written examinations, project. Continuous control

Supervision

Office hours	Yes
Assistants	Yes
Forum	Yes
Others	Office ours by appointment

Resources

Bibliography

Relevant resources (textbook chapters, articles, and videos) posted on moodle page.

CS-410 **Technology ventures in IC**

Cursus	Sem.	Type
Computer science	MA2	Opt.
Data Science	MA2, MA4	Opt.
Mineur STAS Chine	E	Opt.
SC master EPFL	MA2, MA4	Opt.

Language	English
Credits	4
Session	Summer
Semester	Spring
Exam	During the semester
Workload	120h
Weeks	14
Hours	**4 weekly**
Lecture	2 weekly
Project	2 weekly

Remarque

pas donné en 2018-19

Summary

This hands-on class gives graduate students in IC interested in startups the opportunity to learn and put in practice the fundamental skills required to assess a technology concept in the context of a business opportunity. This class is focused only on business opportunities where high-technology

Content

Working in teams, students will learn the fundamentals of:

- *Opportunity assessement*
- *Customer development and validation*
- *Business model alternatives*
- *Intellectual Property*
- *Strategy and Financial planning*
- *Go-to-market, launch, and growth*

This is a hands-on class where students start the class with their own technology venture concept (e.g. the work done as part of their PhD, or some well-formed idea, maybe with a prototype). During the class, they convert their concept into a integrated business plan.

Keywords

Entrepreneurship, startups, technology transfer, intellectual property

Learning Prerequisites

Required courses

- *None – but available to MS and Ph.D. students only*

Learning Outcomes

By the end of the course, the student must be able to:

- Analyze a business plan
- Create a business plan

Teaching methods

- Short ex-cathedra presentations of each topic
- Hands-on seminar with many short student presentations
- Presentations from invited guests, in particluar industry executives and entrepeneurs
- Discussion and case studies

Assessment methods

- In-class participation (30%)
- In-class presentations (30%)
- Final pitch (40%)

Supervision

Office hours	Yes
Assistants	No
Forum	Yes

MATH-342

Time series

Thibaud Emeric Rolland Georges

Cursus	Sem.	Type
Data Science	MA2, MA4	Opt.
Financial engineering	MA2, MA4	Opt.
Mathematics	BA6	Opt.
Mineur STAS Russie	E	Opt.

Language	English
Credits	5
Session	Summer
Semester	Spring
Exam	Written
Workload	150h
Weeks	14
Hours	**4 weekly**
Lecture	2 weekly
Exercises	2 weekly

Summary

A first course in statistical time series analysis and applications, including practical work.

Content

- Motivation; basic ideas; stochastic processes; stationarity; trend and seasonality.
- Autocorrelation and related functions.
- Stationary linear processes: theory and applications.
- ARIMA, SARIMA models and their use in modelling.
- Prediction of stationary processes.
- Spectral representation of a stationary process: theory and applications.
- Financial time series: ARCH, GARCH models.
- State-space models: dynamic linear models, Kalman filter.
- Other topics as time permits.

Learning Prerequisites

Required courses

Probability and Statistics

Recommended courses

Probability and Statistics for mathematicians. A course in linear models would be valuable but is not an essential prerequisite.

Important concepts to start the course

The material from first courses in probability and statistics.

Learning Outcomes

By the end of the course, the student must be able to:

- Recognize when a time series model is appropriate to model dependence
- Manipulate basic mathematical objects associated to time series
- Estimate parameters of basic time series models from data
- Critique the fit of a time series model and propose alternatives
- Formulate time series models appropriate for empirical data
- Distinguish a range of time series models and understand their properties

• Analyze empirical data using time series models

Teaching methods

Ex cathedra lectures, exercises and computer practicals in the R language in the classroom and at home.
Mini-project based on data chosen by the student.

Assessment methods

Mini-project, final exam.
Seconde tentative : Dans le cas de l'art. 3 al. 5 du Règlement de section, l'enseignant décide de la forme de l'examen qu'il communique aux étudiants concernés.

Supervision

Office hours	No
Assistants	Yes
Forum	Yes

Resources

Virtual desktop infrastructure (VDI)

No

Bibliography

A polycopié of the course notes will be available.

Ressources en bibliothèque

• Dynamic Linear Models with R / Petris, Petrone & Campagnoli
• Analysis of Financial Time Series / Tsay
• Introduction to Time Series and Forecasting / Brockwell & Davis
• (electronic version)
• Time Series Analysis and its Applications, with R Examples / Shumway & Stoffer
• (electronic version)
• (electronic version)
• (electronic version)

Notes/Handbook

• Brockwell, P. J. and Davis, R. A. (2016) Introduction to Time Series and Forecasting. Third edition. Springer.
• Shumway, R. H. and Stoffer, D. S. (2011) Time Series Analysis and its Applications, with R Examples. Third edition. Springer.
• Petris, G., Petrone, S. and Campagnoli, P. (2009) Dynamic Linear Models with R. Springer.
• Tsay, R. S. (2010) Analysis of Financial Time Series. Third edition. Wiley.

CS-455 **Topics in theoretical computer science**

Cursus	Sem.	Type
Computer science minor	H	Opt.
Computer science	MA1, MA3	Opt.
Data Science	MA1, MA3	Opt.
SC master EPFL	MA1, MA3	Opt.

Language	English
Credits	4
Session	Winter
Semester	Fall
Exam	During the semester
Workload	120h
Weeks	14
Hours	**4 weekly**
Lecture	3 weekly
Exercises	1 weekly

Remarque

pas donné en 2018-19

Summary

The students gain an in-depth knowledge of several current and emerging areas of theoretical computer science. The course familiarizes them with advanced techniques, and develop an understanding of fundamental questions that underlie some of the key problems of modern computer science.

Content

- Examples of topics to be covered include:

- Streaming: given a large dataset as a stream, how can we approximate its basic properties using a very small memory footprint? Examples that we will cover include statistical problems such as estimating the number of distinct elements in a stream of data items, finding heavy hitters, frequency moments, as well as graphs problems;
- Sketching and sampling: what can we learn about the input from a few carefully designed measurements (i.e. a `sketch') of the input, or just a few samples of the input? We will cover results in sparse recovery and property testing that answer this question for several fundamental problems;
- Sublinear runtime: which problems admit solutions that run faster than it takes to read the entire input? Examples include sublinear time algorithms for graph processing problems, nearest neighbor search and Sparse FFT;
- Communication: how can we design algorithms for modern distributed computation models (e.g. MapReduce) that have low communication requirements? We will discuss graph sketching, a recently developed approach for designing low communication algorithms for processing dynamically changing graphs.

Keywords

streaming, sketching, sparse recovery, sublinear algorithms

Learning Prerequisites

Required courses

Bachelor courses on algorithms, complexity theory, and discrete mathematics.

Learning Outcomes

By the end of the course, the student must be able to:

- Design efficient algorithms for variations of problems discussed in class;
- Analyze formally space/time/communication complexity of randomized algorithms

- Prove space/time/communication lower bounds for variations of problems discussed in class;
- Select appropriately algorithmic tool for big data analysis problem at hand

Teaching methods

Ex cathedra, homeworks, reading

Expected student activities

Attendance at lectures, completing exercises, reading written material

Assessment methods

- Continuous control

Supervision

Office hours	Yes
Assistants	Yes
Others	Electronique forum : Yes

Resources

Bibliography

There is no textbook for the course. Notes will be posted on the course website.

Ressources en bibliothèque

- Randomized Algorithms / Motwani

CS-444 **Virtual reality**

Boulic Ronan

Cursus	Sem.	Type
Computer science	MA2	Opt.
Data Science	MA2, MA4	Opt.
Digital Humanities	MA2	Opt.
SC master EPFL	MA2, MA4	Opt.

Language	English
Credits	4
Session	Summer
Semester	Spring
Exam	During the semester
Workload	120h
Weeks	14
Hours	**3 weekly**
Lecture	2 weekly
Exercises	1 weekly

Summary

The goal of VR is to embed the users in a potentially complex virtual environment while ensuring that they are able to react as if this environment were real. The course provides a human perception-action background and describes the key techniques for achieving efficient VR applications.

Content

The first lectures focus more on the technical means (hw & sw) for acheiving the hands-on sessions:

- Visual display (CAVE and stereoscopy)
- Interaction devices and sensors
- Software environment

The proportion of more theoretical VR and Neuroscience background increases over the semester:

- Key Human perception abilities, Cybersickness, Immersion, presence and flow
- Basic 3D interaction techniques: Magic vs Naturalism
- The perception of action
- Haptic interaction
- What makes a virtual human looking alive ?
- Motion capture for full-body interaction
- VR, cognitive science and true experimental design

Keywords

3D interaction, display, sensors, immersion, presence

Learning Prerequisites

Required courses

(CS 341) Introduction to Computer Graphics

Recommended courses

(CS 211) Introduction to Visual Computing

Important concepts to start the course

from Computer Graphics:
- perspective transformations
- representation of orientation
- 3D modelling hierarchy
- matrix algebra: translation, orientation, composition

Learning Outcomes

By the end of the course, the student must be able to:

- Describe how the human perception-action system is exploited in VR
- Apply the concepts of immersions, presence and flow
- Give an example of applications of VR in different industrial sectors
- Choose a method of immersion suited for a given 3D interaction context
- Explain the possible causes of cybersickness in a given VR system configuration
- Design a VR system involving 3D interactions

Transversal skills

- Set objectives and design an action plan to reach those objectives.
- Assess one's own level of skill acquisition, and plan their on-going learning goals.

Teaching methods

Ex cathedra + Hands-on sessions on VR devices in the first half of the semester,
a mini-project in groups on personal laptops will have to integrate various components of 3D real-time interaction. the group will negociate their project proposal with the course responsible TA who will assess whether it meets the key specifications and is original enough. The proposal can include the use of some VR devices that the IIG research group will lend on a first-come/first-served basis.

Expected student activities

exploit citation analysis tools to evaluate a scientific paper
combine libraries to produce an original 3D interaction
experiment the hands-on practical work in the lab
synthesize the knowledge acquired in course and hands-on in the quizzes and final oral

Assessment methods

Throughout semester: 4 Hand-on sessions (4%), 2 Quizzes (10%), 1 paper citation study (16%), 1 mini-project (40%), 1 oral (30%)

Supervision

Office hours	No
Assistants	Yes
Forum	Yes

Resources

Virtual desktop infrastructure (VDI)

No

Bibliography

- Course notes will be updated and made available after each course, with links to key sites and on-line documents
- J. Jerald, The VR Book, ACM Press 2015
- Parisi, Learning Virtual Reality, O'Reilly 2015
- Le Traité de Réalité Virtuelle (5 vol.) Presses des Mines, ParisTech, 2006-2009, available on-line, free for student upon registration.
- Doug A. Bowman, Ernst Kruijff, Joseph J. LaViola, and Ivan Poupyrev. 2004. 3D User Interfaces: Theory and Practice. Addison Wesley Longman Publishing Co., Inc., Redwood City, CA, USA.

Ressources en bibliothèque

- 3D User Interfaces: Theory and Practice / Bowman
- Learning Virtual Reality / Parisi
- The VR Book / Jerald
- Le Traité de Réalité Virtuelle / Fuchs

Notes/Handbook

pdf of slides are made visible after the ex-cathedra courses

Websites

- http://www.thevrbook.net/

INDEX PAR COURS

Cours pas donnés en 2018-2019

INDEX PAR ENSEIGNANTS

INDEX PAR ENSEIGNANTS