

2I013 Groupe 1

Projet Foot

Maxime Sangnier – Nicolas Baskiotis

maxime.sangnier@lip6.fr

2019

Laboratoire d'Informatique de Paris 6 (LIP6)

Sorbonne Université

Outline

Séance précédente

Apprentissage artificiel (Machine learning)

Les problématiques générales

Exemple de code

Objectifs du TME

Séance précédente

Bilan du TME

Vous avez :

- paramétré les stratégies (tir, défense, anticipation de la position de la balle...);
- optimisé les stratégies par recherche en grille ou algorithme génétique;
- implémenté des stratégies pour le 2v2 et le 4v4.

Apprentissage artificiel (Machine learning)

Quelques définitions

Scott, 1983

Learning is the organization of experience.

Herbert Simon, 1983

Learning is any change in a system that allows it to perform better the second time on repetition of the same task or another task drawn from the same population.

Marvin Minsky, 1985

Learning is making useful changes in mind.

Ryszard Michalski, 1986

Learning is constructing or modifying representations of what is being experienced.

L'apprentissage artificiel :

- étudie les algorithmes qui améliorent leur performance sur une tâche donnée en fonction de leur expérience.
- fondements mathématiques, informatiques et applications concrètes des systèmes qui apprennent, raisonnent et agissent.

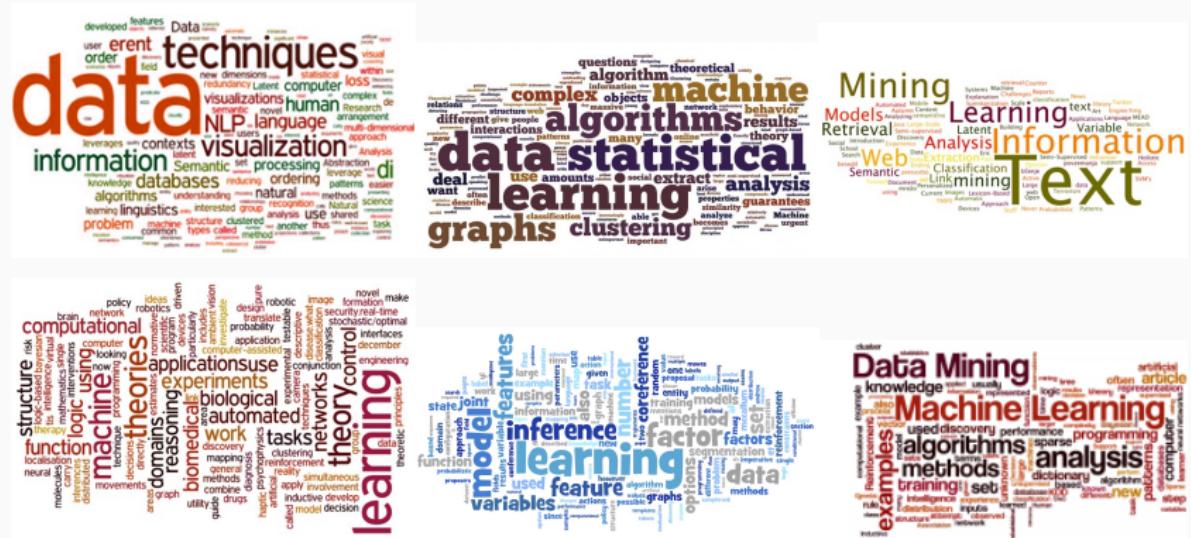
Touche à beaucoup de domaine, interdisciplinarité très forte

- Statistiques : théorie de l'apprentissage, fouille de données, inférence
- Informatique : IA, vision, RI
- Ingénierie : signal, contrôle, robotique
- Science cognitive, psychologie, neuroscience, épistémologie
- Économie : théorie de la décision, théorie des jeux

En quelques mots

- Prédire une quantité à partir d'une observation.
- Comprendre des phénomènes physiques grâce à un ensemble d'observations.

Apprentissage artificielle ? (Machine learning)



A votre avis, ça regroupe quoi ?

Classification de documents

E-mails en spam, shopping, travail, ...

Supprimer tous les spams maintenant (les messages se trouvant dans le dossier Spam depuis plus de 30 jours sont automatiquement supprimés)		
<input type="checkbox"/>	Tatianna	Re: Para os homens - Vai lhe interessar muito!
<input type="checkbox"/>	comebuy	Téléphones les plus compétitifs de Comebuy
<input type="checkbox"/>	Francois	100 raisons de jouer sur Majestic
<input type="checkbox"/>	Fund Investigation Bureau	TREAT AS URGENT RIGHT AWAY
<input type="checkbox"/>	Mrs Elizabeth Johnson	Hello My Beloved One.
<input type="checkbox"/>	Evellyn	Re: Amigo, não está satisfeito com o tamanho? Isto pode te ajudar!
<input type="checkbox"/>	Amanda, Amanda (2)	Re: Amigo, o que vc faria com 10cm a mais?
<input type="checkbox"/>	Groupe Partouche	Et encore un gagnant au Megapot !
<input type="checkbox"/>	Carli, Joshua Daniel	N/A
<input type="checkbox"/>	RCH Tournoi	Votre Semaine avec 100000 en Tout
<input type="checkbox"/>	Jemmy Klamet	Nicolas Baskiotis F-E..E..L-I..N G .. H_O..R_N-Y?-_-G-E-T _L_A_I_D --N_O_W !
<input type="checkbox"/>	Jean-Pierre	Les meilleurs casinos pour les joueurs français

gmail.com

Principale	Réseaux sociaux	Promotions
	CollierPrenom Annonce Spécial St Valentin - 3 Jours Seulement - 15% de Réduction !	
	SoftLayer.com Annonce Get a Secure Cloud - We've secured the public cloud with private servers, private networks, and full private clouds.	
<input type="checkbox"/>	Booking.com Last-minute deals for Montréal and London. Get them before they're gone!	28/12/2014
<input type="checkbox"/>	Voyages-snfc.com DERNIERE MINUTE NOUVEL AN : profitez des meilleurs prix !	26/12/2014
<input type="checkbox"/>	Impossible Year's End Clearance – Up to 20% off Film and Accessories	26/12/2014
<input type="checkbox"/>	Booking.com Nicolas – you qualify for at least 20% off places to stay	26/12/2014
<input type="checkbox"/>	Communauté d'entraide Gr. Nicolas, des questions sur vos produits ?	25/12/2014

En texte toujours

Reconnaissance de chiffres

8 2 9 4 4 6 4 9 7 0 9 2 9 5 1 5 9 1 0 3
2 3 5 9 1 7 6 2 8 2 2 5 0 7 4 9 7 8 3 2
1 1 8 3 6 1 0 3 1 0 0 1 1 2 7 3 0 4 6 5
2 6 4 7 1 8 9 9 3 0 7 1 0 2 0 3 5 4 6 5

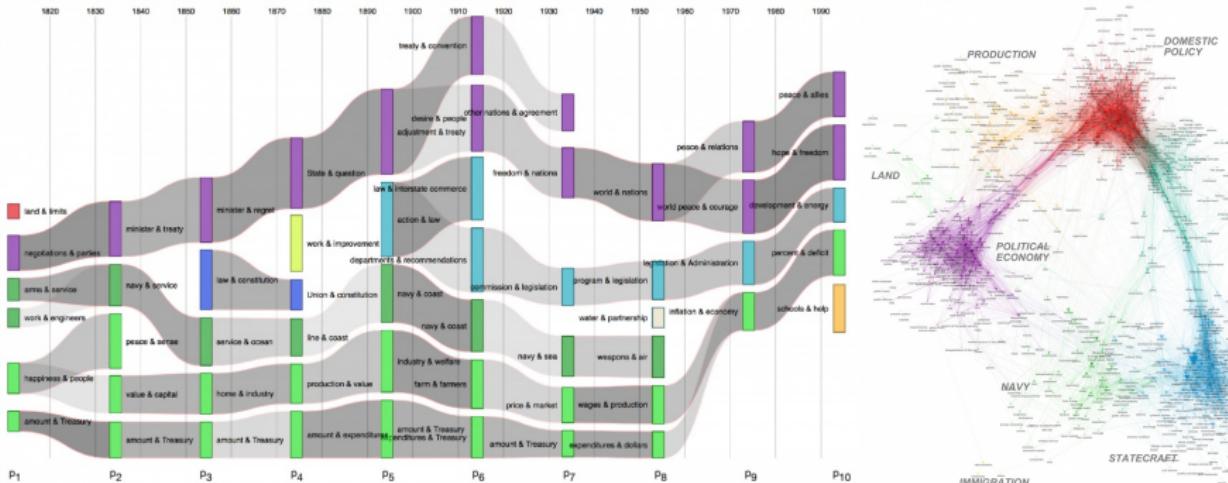
Ou de captcha

[Yann et al. 08], Newcastle University



Sur des documents

Détection de thèmes (topic detection)



Analyse de 255 discours de l'état de l'union, États-Unis

[Rule et al, 2014]

Et plein d'autres applications : traduction, détection de plagiat, résumé automatique, ...

En image

Détection de visages

(opencv)



Mais aussi ...

(betafaceapi.com)

	Score: 0.42 X: 398.67 Y: 29.66 Width: 26.79 Height: 26.79 Angle: -5.45	age : 37 (16%), gender : male, race : white, chin size : average, color background : 4c5042 (15%), color clothes middle : 3295eb (48%), color clothes sides : 38a9f5 (96%), color eyes : ac8066, color hair : fbf2ea (80%), color mustache : a56855 (65%), color skin : dbb5a1, eyebrows corners : extra low, eyebrows position : average, eyebrows size : extra thin, eyes corners : low, eyes distance : average, eyes position : average, eyes shape : extra round, glasses rim : no, hair beard : none, hair color type : blond (80%), hair forehead : yes, hair length : none, hair mustache : thick, hair sides : very thin, hair top : short, head shape : average, head width : extra narrow, mouth corners : low, mouth height : extra thin, mouth width : extra small, nose shape : extra straight, nose width : wide, teeth visible : no [collapse]
	Score: 0.57 X: 216.66 Y: 155.08 Width: 28.34 Height: 28.34 Angle: 0.95	age : 46 (23%), gender : male, race : white, chin size : extra small, color background : 0c0cd (36%), color beard : 4a2617 (50%), color clothes middle : a22e55 (82%), color clothes sides : a54031 (74%), color eyes : 966a58, color hair : 655348 (77%) , color skin : b98f78, eyebrows corners : average, eyebrows position : extra high, eyebrows size : extra thin, eyes corners : average, eyes distance : close, eyes position : extra low, eyes shape : extra thin, glasses rim : no, hair beard : short, hair color type : brown light (77%), hair forehead : no, hair length : short, hair mustache : none, hair sides : thin, hair top : short, head shape : rect, head width : extra wide, mouth corners : average, mouth height : extra thin, mouth width : average, nose shape : average, nose width : extra narrow, teeth visible : no [collapse]

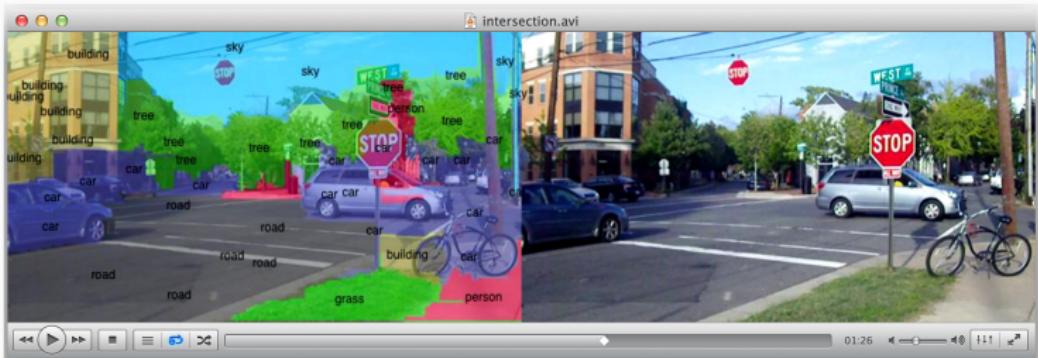
Classification et organisation automatique



En image

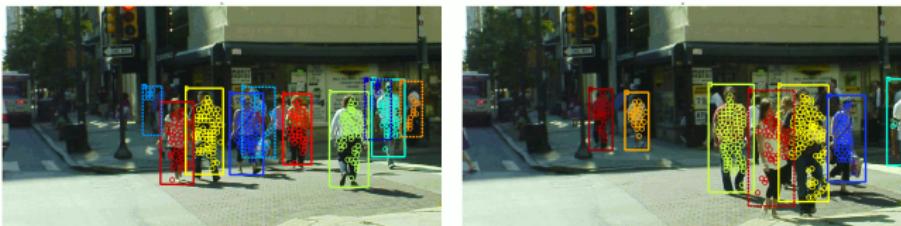
Détection d'objets

teradeep.com, Purdue University



Tracking

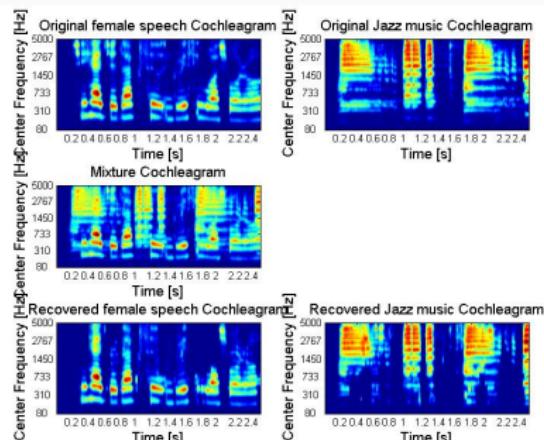
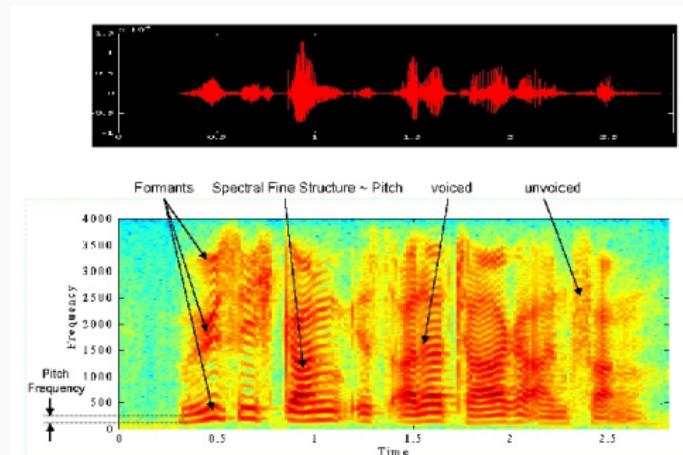
[Fragkiadaki et al. 12], Pennsylvania University



Et l'audio ...

Reconnaissance de la parole, séparation de sources

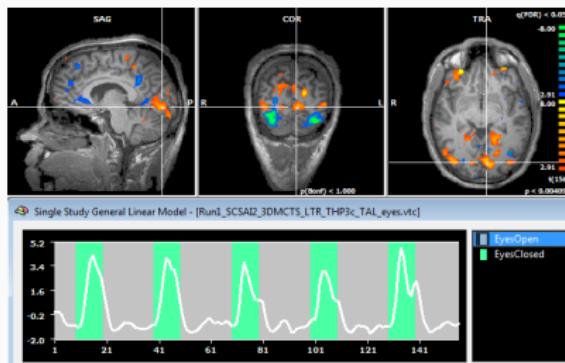
<http://markus-hauenstein.de>



Mais aussi débruitage, transcription musicale, reconnaissance du locuteur, classification/identification de musiques ...

Interface cerveau-machine (BCI)

Classification d'actions, de pensées



Contrôle



Objets connectés

Traqueurs d'activité



Surveillance vidéo, monitoring consommation électrique, sécurité réseau

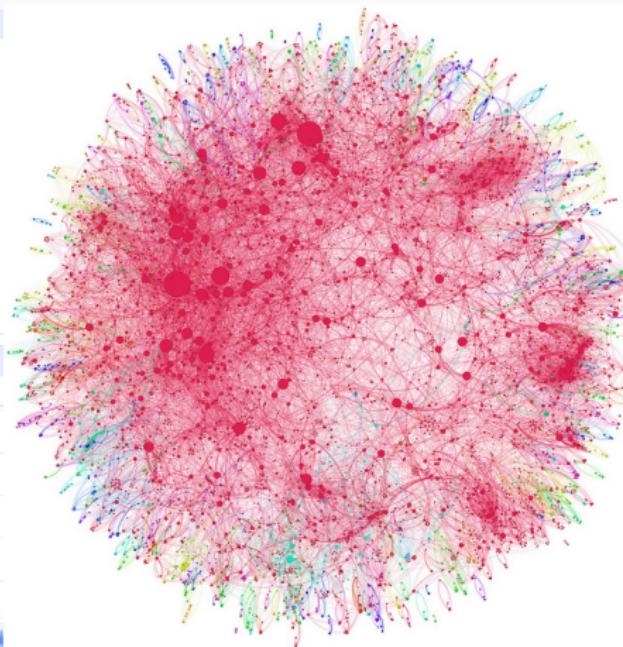


Détection de communauté, phénomènes de diffusions, classification

Largest Diffusion Network



Meme Activity



Matchmaking

de profils, sites de rencontre



Experts, CV - Emplois, Jeux



LinkedIn
viadeo



Systèmes de recommandation

De musiques, de films, de produits, d'amis

Recommendation Engine Home Item Store My Items Hello cincodosis@gmail.com Log off

Similar Artists



1 Bob Dylan
2 Radiohead ←
3 Led Zeppelin
4 The Rolling Stones
5 Pink Floyd
6 David Bowie
7 The Who
8 John Lennon

Movies Music Articles Artists

Search All

Umbrellas of Cherbourg, The Jacques Demy Drama | 1964 | Unrated Select

Broadway Palace Jonathan Kaplan Drama | 1999 | PG-13 Select

West Berlin Zadie Smith Drama | 1998 | PG-13 Select

Suspect Peter Yates Thriller | 1987 | R Select

Heights Jeremy Kagan Drama | 2004 | R Select

Babylon A.D. Mathieu Kassovitz Action | 2008 | PG-13 Select

amazon.com

Recommended for You

Amazon.com has new recommendations for you based on items you purchased or told us you own.

The Little Big Things: 162 Ways to Pursue EXCELLENCE

Fascinate Your 7 Triggers to Persuasion and Captivation

Sherlock Holmes [Blu-ray]

Alice in Wonderland [Blu-ray]

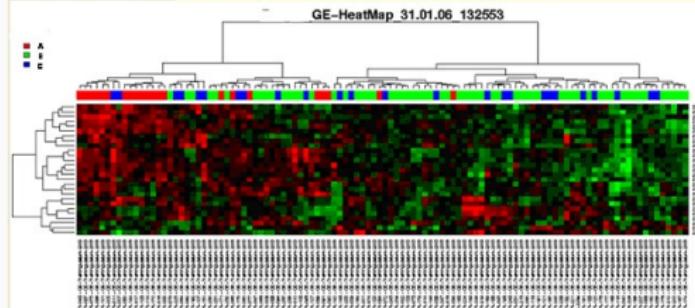


Et dans d'autres sciences

Biologie IZBI, Leipzig University

Gene Signal Value Visualization - Gene Expression Heatmap

This form draws the heatmap of Gene Expression signals determined by a selected Experiment Group and a selected Gene Group.



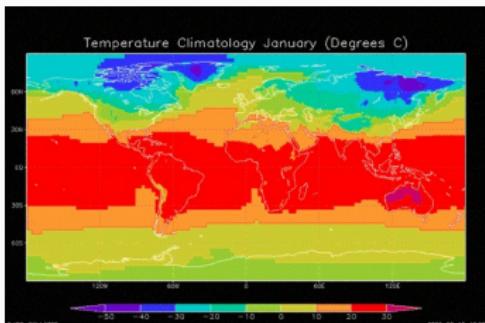
Économie



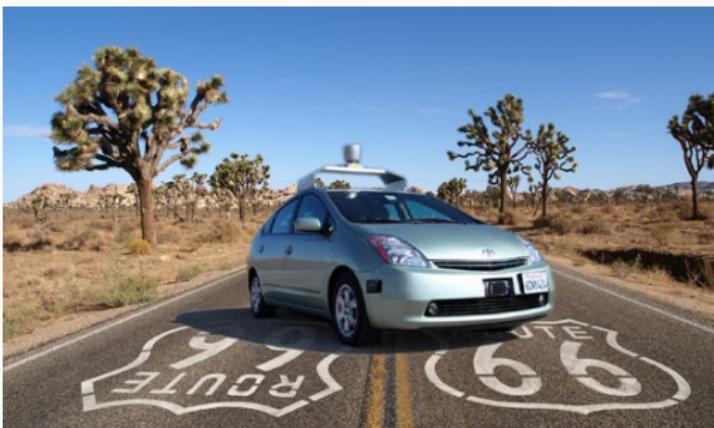
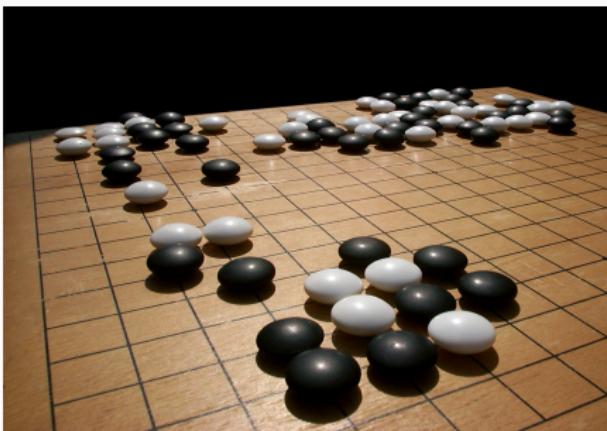
Astronomie



Climatologie (complémentation données)

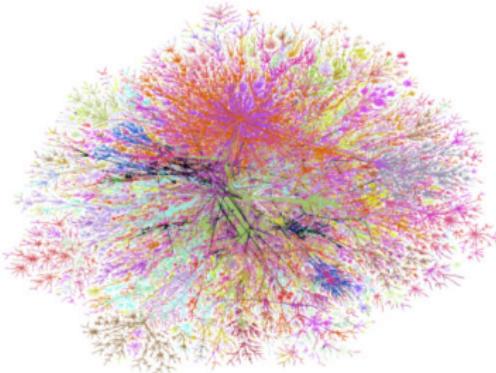


Dans les jeux et la robotique



L'apprentissage aujourd'hui : Big Data

- Webpages (content, graph)
- Clicks (ad, page, social)
- Users (OpenID, FB Connect)
- e-mails (Hotmail, Y!Mail, Gmail)
- Photos, Movies (Flickr, YouTube, Vimeo ...)
- Cookies / tracking info (see Ghostery)
- Installed apps (Android market etc.)
- Location (Latitude, Loopt, Foursquared)
- User generated content (Wikipedia & co)
- Ads (display, text, DoubleClick, Yahoo)
- Comments (Disqus, Facebook)
- Reviews (Yelp, Y!Local)
- Third party features (e.g. Experian)
- Social connections (LinkedIn, Facebook)
- Purchase decisions (Netflix, Amazon)
- Instant Messages (YIM, Skype, Gtalk)
- Search terms (Google, Bing)
- Timestamp (everything)
- News articles (BBC, NYTimes, Y!News)
- Blog posts (Tumblr, Wordpress)
- Microblogs (Twitter, Jaiku, Meme)



>10B useful webpages

Carnegie Mellon University

extrait du cours d'A.

Smola

Entreprises concernées :

Yahoo, Google, Amazon, Netflix, Microsoft, Apple, Xerox, Samsung, Critéo, Facebook, Twitter, Flickr, Instagram, Reddit, Valve, Steam, Deezer, Dailymotion, Youtube, STIF, SNCF, AXA, EDF, GDF-Suez, Veolia, Safran, Thalès, les médias, ...

Les masters ML à SU

- Master d'informatique, spécialité DAC : M1, M2.
<http://dac.lip6.fr/master/>
- Master maths-info M2A : M2 après M1 info ou M1 maths.
<https://m2a.lip6.fr/>

Les problématiques générales

Données du problème

- Une représentation X des objets de l'étude
- Une sortie d'intérêt y qui peut être numérique, catégorielle, structurée, complexe (label, réponse, étiquette, ...)
- Un ensemble d'exemples, d'échantillons, sous leur représentation X et avec leur sortie connue $\{(x_1, y_1), \dots, (x_n, y_n)\}$

Objectifs

- Prédire de manière précise la sortie y pour un nouvel exemple x non vu
- Comprendre quels facteurs influencent la sortie
- Évaluer la qualité de nos prédictions

Apprentissage non supervisé

Données du problème

- Une représentation X des objets de l'étude
- Un ensemble d'exemples, d'échantillons, sous leur représentation X ,
 $\{x_1, \dots, x_n\}$
- Pas de variable de sortie !

Objectifs

- Trouver des groupes d'objets “semblables”
- Organiser les données d'une manière “logique”
- Trouver les “similarités” des objets
- Trouver des “représentations” des objets

Apprentissage par renforcement

Apprentissage continu en fonction du retour d'expérience

Données du problème

- Un état décrit l'environnement courant
- Un ensemble d'actions sont possibles
- Une politique permet de choisir en fonction de l'état l'action à effectuer
- A l'issue de chaque action, une récompense est observée

Objectifs

- S'améliorer ! (améliorer la politique de choix de l'action)
- Éviter les situations d'échecs
- Comprendre la dynamique du problème

Les grandes familles

Apprentissage supervisé

- Classification
- Régression
- Forecasting
- Compléction de données
- Ranking
- Recommandation

Apprentissage non supervisé

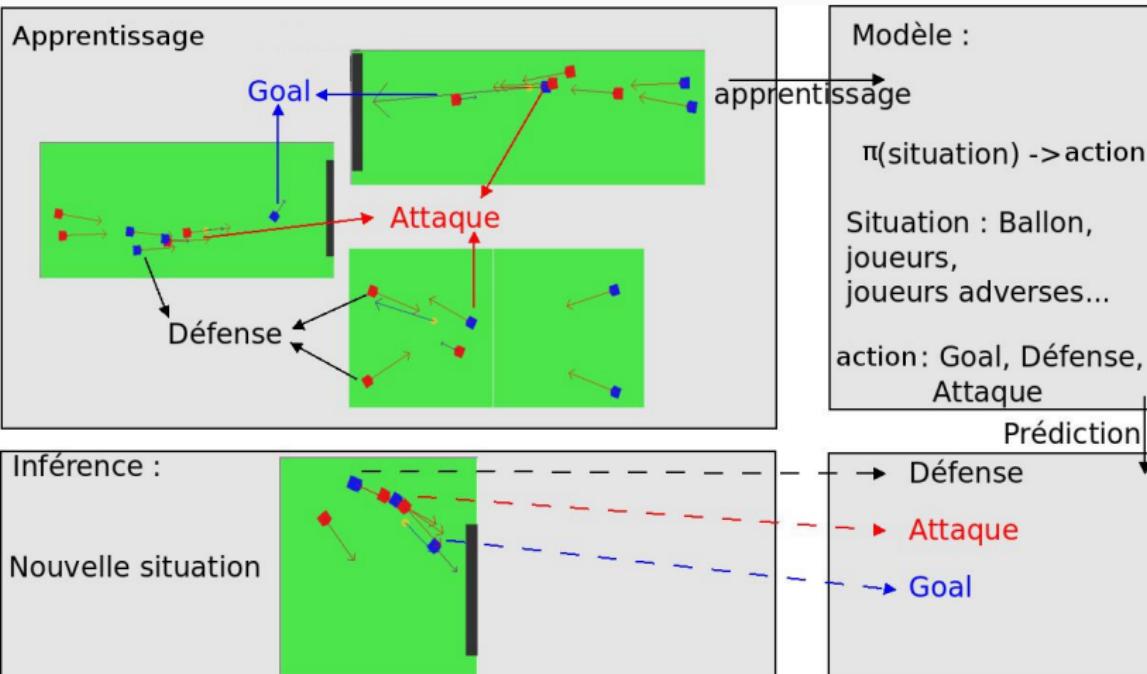
- Clustering
- Apprentissage de représentation, de dictionnaire
- Analyse de séquences
- Représentation hiérarchique
- Détection d'anomalies

Apprentissage par renforcement

- Apprendre à jouer
- Apprendre à interagir avec l'environnement

Et l'utilité dans le cadre du projet ?

Apprentissage par renforcement



Principe

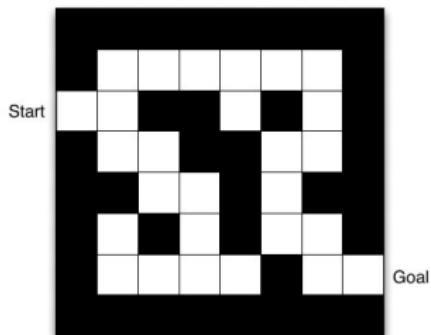
- apprendre une *politique* pour réagir à l'environnement
- ⇒ une fonction $\pi(s)$ renvoyant pour chaque état une action ou une distribution de probabilité sur les actions
- Objectif : trouver la politique qui maximise l'espérance des récompenses

Notion fondamentale : fonctions (équivalentes)

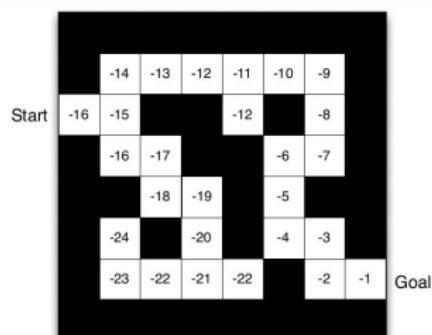
- de valeur d'état $v_\pi(s) = \mathbb{E}(r_{t+1} + \gamma r_{t+2} + \gamma^3 r_{t+3} + \dots + \gamma^T r_T | s_t = s)$ prédit le score d'un état
- de valeur d'état-action (ou de qualité)
 $Q_\pi(s, a) = r_t + \gamma \sum_{s'} \mathbb{P}(s'|s, a) v_\pi(s)$ prédit le score d'une action entreprise dans un état donné.

Exemple : labyrinthe

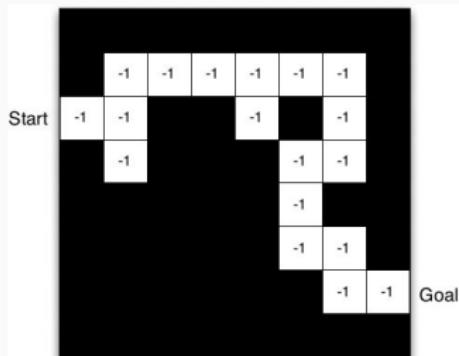
environnement



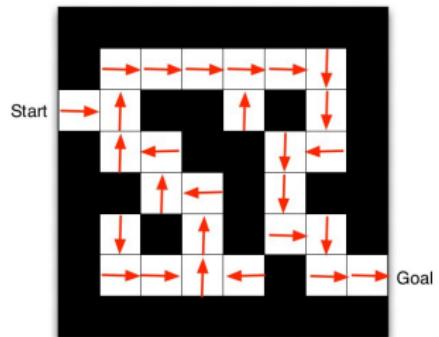
valeur d'état



récompense



meilleur action



Apprentissage

- Équation de Bellman : $Q_\pi(s, a) = r_t + \gamma \max_{a'} Q_\pi(s, a')$
- r_t : récompense immédiate
- $\gamma \max_{a'} Q_\pi(s, a')$: récompense future
- $\gamma \in [0, 1]$: facteur d'actualisation
- Simulations numériques (beaucoup)

Exploration vs exploitation : essai-erreur

- Au tout début, on ne connaît rien !
- la politique initiale : au hasard
- Besoin de tester de nouveaux choix pour certains états, voir si ça améliore le score global \Rightarrow Exploration
- Mais pas trop ! se guider des bons choix actuels \Rightarrow Exploitation

Q-learning

Initialize $Q(s, a), \forall s \in \mathcal{S}, a \in \mathcal{A}(s)$, arbitrarily, and $Q(\text{terminal-state}, \cdot) = 0$
Repeat (for each episode):

 Initialize S

 Repeat (for each step of episode):

 Choose A from S using policy derived from Q (e.g., ε -greedy)

 Take action A , observe R, S'

$Q(S, A) \leftarrow Q(S, A) + \alpha [R + \gamma \max_a Q(S', a) - Q(S, A)]$

$S \leftarrow S'$;

 until S is terminal

Exemple de code

Exemple de code

```
from socceria import QLearning, RandomPos, QStrategy

# Strategy
QTestStrategy = QStrategy()
QTestStrategy.add('right', SimpleStrategy(shoot_right, ''))
QTestStrategy.add('left', SimpleStrategy(shoot_left, ''))
QTestStrategy.add('up', SimpleStrategy(shoot_up, ''))
QTestStrategy.add('down', SimpleStrategy(shoot_down, ''))

# Learning
expe = QLearning(strategy=QTestStrategy, monte_carlo=False)
expe.start(fps=1500)

with open('qstrategy.pkl', 'wb') as fo:
    QTestStrategy.qtable = expe.qtable
    pkl.dump(QTestStrategy, fo)

# Test
with open('qstrategy.pkl.351', 'rb') as fi:
    QTestStrategy = pkl.load(fi)

# Simulate and display the match
simu = RandomPos(QStrategy)
simu.start()
```

Résultat pour le tableau état-action

RIGHT	DOWN	DOWN
RIGHT	DOWN	DOWN
RIGHT	RIGHT	RIGHT
RIGHT	UP	UP
RIGHT	UP	UP

La classe QLearning

```
class QLearning(object):
    def __init__(self, strategy, simu=None, max_steps=1000000,
                 max_round_step=100, monte_carlo=False):
        self.strategy = strategy
        self.simu = simu
        self.max_steps = max_steps
        self.max_round_step = max_round_step
        self.monte_carlo = monte_carlo

    def start(self, show=True, fps=None):
        if not self.simu:
            team1 = SoccerTeam("Team\u22251")
            team2 = SoccerTeam("Team\u22252")
            team1.add(self.strategy.name, self.strategy)
            team2.add(Strategy().name, Strategy())
            self.simu = Simulation(team1, team2, max_steps=self.max_steps)
            self.simu.listeners += self

        if show:
            show_simu(self.simu, fps=fps)
        else:
            self.simu.start()

    def get_res(self):
        return self.qtable
```

Début de match

```
def begin_match(self, team1, team2, state):
    self.last_step = 0 # Step of the last round
    self.qtable = dict() # Q table

def begin_round(self, team1, team2, state):
    ball = Vector2D.create_random(low=0, high=1)
    ball.x *= GAME_WIDTH
    ball.y *= GAME_HEIGHT

    # Player and ball position (random)
    self.simu.state.states[(1, 0)].position = ball.copy() # Player position
    self.simu.state.states[(1, 0)].vitesse = Vector2D() # Player acceleration
    self.simu.state.ball.position = ball.copy() # Ball position

    # Last step of the game
    self.last_step = self.simu.step
    self.last_state = None
    self.last_score = self.simu.score[1] # Score of Team 1
    self.cur_state = self.strategy.get_state(state, id_team=1, id_player=0)
    self.rewards = []
```

À chaque pas de temps

```
def update_round(self, team1, team2, state):
    # Q-learning update
    self.qupdate(state)

    if state.step > self.last_step + self.max_round_step:
        # Change action when state doesn't change
        if self.cur_state == self.last_state:
            self.strategy.strategy = self.next_action(self.cur_state)
    self.last_state = self.cur_state
```

MAJ du tableau état-action

```
def qupdate(self, state):
    qstate_next = self.strategy.get_state(state, id_team=1, id_player=0)
    if self.cur_state != qstate_next:
        qaction = self.strategy.strategy # Strategy name
        key = (self.cur_state, qaction)

        # Future Q-value
        qvalues_next = [q for k, q in self.qtable.items() \
                        if k[0] == qstate_next]
        qnext = max(qvalues_next, default=0)

        # Reinforcement
        score = state.score[1]
        if score > self.last_score:
            self.last_score = score
            r = 0
        else:
            r = -1 if state.goal == 0 else -10
        qvalue = r + 0.9 * qnext
```

MAJ du tableau état-action

```
# Update Q-table
self.rewards.append((key, r))
if not self.monte_carlo:
    if key in self.qtable:
        self.qtable[key] = 0.5 * self.qtable[key] + 0.5 * qvalue
    else:
        self.qtable[key] = qvalue

# Change action
self.strategy.strategy = self.next_action(qstate_next)

# Update current state
self.cur_state = qstate_next
```

Prochaine action

```
def next_action(self, qstate_next):
    minq = min([q for key, q in self.qtable.items() \
                if key[0] == qstate_next], default=0)
    minq = min(minq, 0)

    prob = [0.1 if (qstate_next, name) not in self.qtable \
            else self.qtable[(qstate_next, name)] - minq + 0.1 \
            for name in self.strategy.strategy_names]
    prob = np.asarray(prob)
    if prob.sum() < 1e-15:
        prob = None
    else:
        prob /= prob.sum()

    return choice(list(self.strategy.strategy_names), p=prob)
```

La classe QStrategy

```
class QStrategy(Strategy):
    def __init__(self):
        Strategy.__init__(self, "Q-learning")
        self.strategies = dict()
        self.current_strategy = None
        self.qtable = None

    def add(self, name, strategy):
        self.strategies[name] = strategy
        if not self.current_strategy:
            self.current_strategy = name

    def get_state(self, state, id_team, id_player):
        s = SuperState(state, id_team, id_player)
        x = int(s.player.x / GAME_WIDTH * 3)
        y = int(s.player.y / GAME_HEIGHT * 5)
        return x, y, state.goal > 0
```

La classe QStrategy

```
def compute_strategy(self, state, id_team, id_player):
    if self.qtable is None:
        strat = self.strategies[self.current_strategy]
        return strat.compute_strategy(state, id_team, id_player)
    else:
        qstate = self.get_state(state, id_team, id_player)
        strategy = max([(q, key[1]) for key, q in self.qtable.items() \
                        if key[0]==qstate], default=(None, None))[1]
        if strategy is not None:
            strat = self.strategies[strategy]
            return strat.compute_strategy(state, id_team, id_player)
        else:
            return SoccerAction()

@property
def strategy_names(self):
    return self.strategies.keys()

@property
def strategy(self):
    return self.current_strategy

@strategy.setter
def strategy(self, name):
    self.current_strategy = name
```

Objectifs du TME

Objectifs du TME

Objectifs du TME

- définir des stratégies élémentaires (défense, attaque, attente de balle...);
- construire des stratégies complexes (4v4) à partir de ces stratégies simples;
- construire automatiquement ces stratégies complexes par apprentissage automatique.

A chaque TME

Mettre à jour le dépôt contenant le simulateur :

```
cd [DOSSIER DU SIMULATEUR]  
git pull
```