

USING MACHINE LEARNING TO ASSESS PLAYER QUALITY IN TEAM SPORTS

Albrecht Zimmermann



GREYC

Laboratoire de recherche en sciences du numérique



Normandie Université



Assessing players a worth-while challenge in team sports

1. Is she worth a large contract ?
 - Who's the best player in the NHL and are they undervalued ?
2. How do we find a (cheaper) replacement ?
 - Replacing CR7, for example.
3. Do they perform well in particular situations?
 - Does Neymar make good decisions under pressure ?
4. Can we find hidden gems ?
 - Who's not in the top-10 but should be ?

How to assess a player ?

1. Does things that help the team, avoids things that hurt it
 - Scoring helps
 - Preventing to score helps
 - Putting teammates in position to score helps
 - Losing the ball doesn't
2. Standard human way of assessment : discrete actions
 - Goal
 - Assist
 - Block

Harder for more diffuse actions :

- Setting a screen
- Help defense

How to assess the quality of an action

- Human experts
 - Biased
 - Slow (90 mins+)
 - Not that many experts around
- Hard to do for actions far in the past
 - The pass directly before a goal is clear
 - The screen 10 seconds before, though ?

Assessment via machine learning

- Learn a model that assigns (numerical) values to actions
 - Ground truth : based on expert assessments
 - Or on back-propagation of positive/negative outcomes
- Score all of a player's actions
 - Even the ones that occurred long before a score
 - Or the ones that occurred off the ball
- Calculate (category-specific) averages
 - Some player are better passers/defenders etc

High-level plan

- Soccer :
 - « Not all passes are created equal », Power et al. KDD '17
 - « Actions speak louder than goals », Decroos et al. KDD '19
 - « Choke or shine », Bransen et al. MSSAC '19
- Basketball :
 - « Deephoops: Evaluating micro-actions in basketball using deep feature representations of spatio-temporal data », Sicilia et al. KDD '19
 - « Bhostgusters: Realtime interactive play sketching with synthesized NBA defenses », Seidl et al. MSSAC '18
- Ice hockey :
 - « Deep Reinforcement Learning in Ice Hockey for Context-Aware Player Evaluation », Liu & Schulte IJCAI '18

Not all passes are created equal

- Consider passes' risk and reward
- Event data + player tracking data
 - Derived features
 - Tactical features
 - Formation features
- **Logistic regression** to learn risk
 - Wanted interpretability
- Pass reward : probability of score within 10 secs
 - Also logistic regression
- Formations
 - **Clustering**

Actions speak louder than goals

- Assess all actions, not just passes
- Event stream data
 - SPADL, derived + contextual
- Valuing Actions by Estimating Probabilities (VAEP)
 - Change in probability for scoring/conceding in « near future »
- **CatBoost** to learn probabilities for given game state
 - Form of [gradient boosted decision trees](#)

Choke or shine

- Do players make good choices/execute well under pressure ?
- Pre-game (mental) pressure, in-game (mental) pressure
- Pre-game :
 - Team ambition : **k-means clustering**
 - Pair-wise rankings of games : **Gradient-boosted tree** to learn pressure
- In-game :
 - Based on state t , predict goals until end of match
 - **Auto-Differentiation Variational Inference** algo for learning
- « Actions speak louder than goals » values
- **Gradient-boosted trees** learn success/failure of actions

Deephoops

- « Micro-actions » : screens, cuts etc
- Player-tracking data
 - Sliding window instead of full possessions
- Labeled windows → expected points value → change in EPV
- **Long short-term memory** (LSTM) model to predict
 - Complex architecture
- Combined w/learned player **embedding**
- « What if » assessments

Bhostgusters

- How would the defending team respond ?
- Player tracking data
 - + game context data
- Learn **typical** player trajectories by **LSTM** model
- Evaluate observed offensive possessions
 - Actions can be switched out for « what if » scenarios
- Sketch hypothetical ones
 - → routes for players, i.e. succession of actions : pass, dribble, move etc

Deep **Reinforcement Learning** in Ice Hockey for Context-Aware Player Evaluation

- How good is the action a player performs in a particular situation ?
- Event data
 - + derived features
- Split up into goal-scoring episodes
 - Subdivided via possession changes
- Learn Q-function to assess actions
- **LSTM network** as learner
- Changes in Q value used to assess player

Let's talk data – what is event stream data ?

1. Something discrete happens

- Pass, tackle, block, turnover etc
- Events = actions

2. Annotated with

- x, y coordinates (origin, arrival)
- Involved player(s)
- Time stamp (beginning, end)

3. Derived from annotators' work

- Watch the video, use particular software
- Some work on automatizing things

4. Irregular intervals

• « Simple pass »

- PlayerID : 25413
- x:49, y:49
- x:78, y:31
- EventSec :2.7586489999999912

• « High pass »

- PlayerID : 370224
- x:78, y:31
- x:75, y :51
- EventSec :4.9468500000000012

Pappalardo, Luca; Massucco, Emanuele (2019): Soccer match event dataset. figshare. Collection.
<https://doi.org/10.6084/m9.figshare.c.4415000.v5>

Let's talk data – what is (player) tracking data ?

1. Where are the players ?
 - x, y coordinates
2. Where's the ball ?
 - x, y, z coordinates
3. Derived from camera images + entity recognition
 - Team shirt colors help
4. Regular intervals
 - Typically 20-25 Htz



Let's talk data

- Why events data ?
 - Because events/actions are what makes the game go 'round
- Why not only events data ?
 - Because a pass w/o close-by defenders is a completely different thing
- Why tracking data ?
 - Because it gives a complete picture
- Why not only tracking data ?
 - Because positions don't tell anything about actions

Not all passes are created equal : Objectively Measuring the Risk and Reward of Passes in Soccer from Tracking Data

- Pass the **most** often occurring action in soccer
- Some easy ways of assessing passes :
 - Success/failure
 - Assist
- Assist = great pass
- Successful pass = good pass ?
- Failed pass = bad pass ?

	High risk	Low risk
High reward	?	Good pass
Low reward	Bad pass	?

Not all passes : data (STATS Inc paper)

- 726 EPL matches
- Player positions sampled at 10hz
- Event data, e.g. pass, tackle, shot :
 - x, y ball position, time stamp
 - Player in position
 - Player identity of opposing player for duels
- 571,287 passes, of which 468,265 successful (84.35%)

Sideways, backwards more successful
Closer to goal less successful
Longer less successful

Imbalance problematic

Pass risk

- Probability that a player completes a pass he plays
- Derived features :
 - Speed of passer and receiver
 - Speed/distance of nearest defenders
 - Nearest defender angle to passing line
 - First time pass ?
 - Time from regaining possession
 - Most likely intended receiver : $\text{distance}/\text{min distance} * \text{angle}/\text{min angle}$
- Risk learned via **logistic regression**

Data transformation

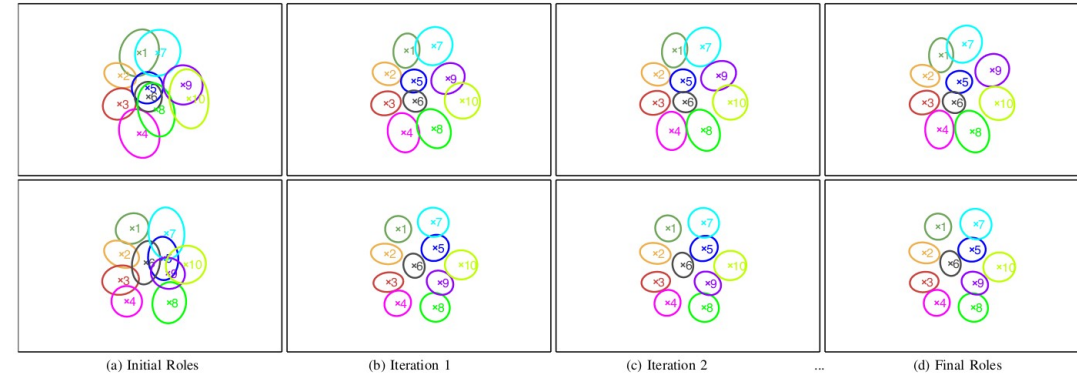
Tactical + formation features

- Build-up, counter-attack, unstructured play
 - Expert assessment
- Formation **clustering**
 - Polar coordinates of all players w.r.t. goal
 - Polar coordinates of defenders w.r.t. passer/receiver
- **K-means** to identify high-, medium-, low-block (defensive)

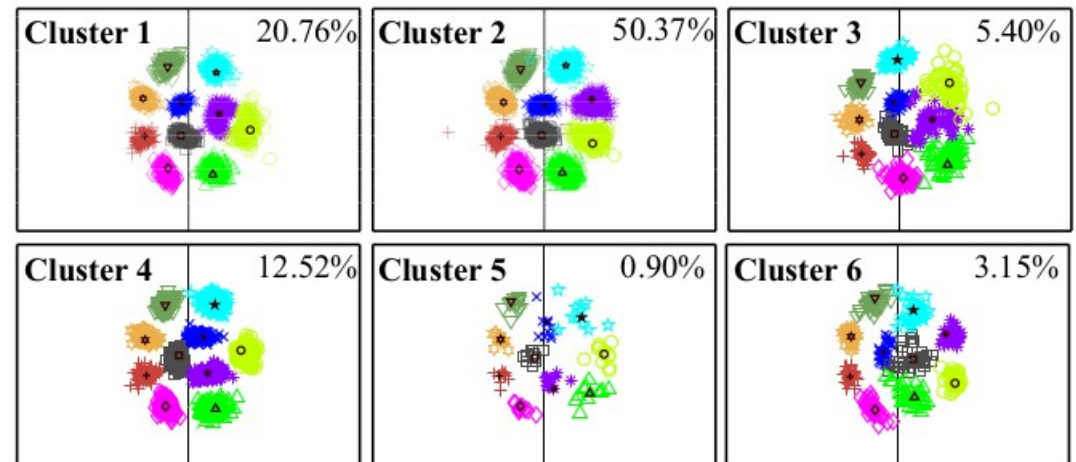
Data transformation/
augmentation

Formation clustering

- Group similar data points, separate dissimilar ones
 - Unsupervised problem
- Team formation = set of roles
- One team per frame
- Normalized → zero mean for all tracking data
- Cluster « role positions » until convergence → formation
 - EM algorithm



- Results in 1411 formations
 - Cluster with HAC using Earth Mover's distance → 6 clusters

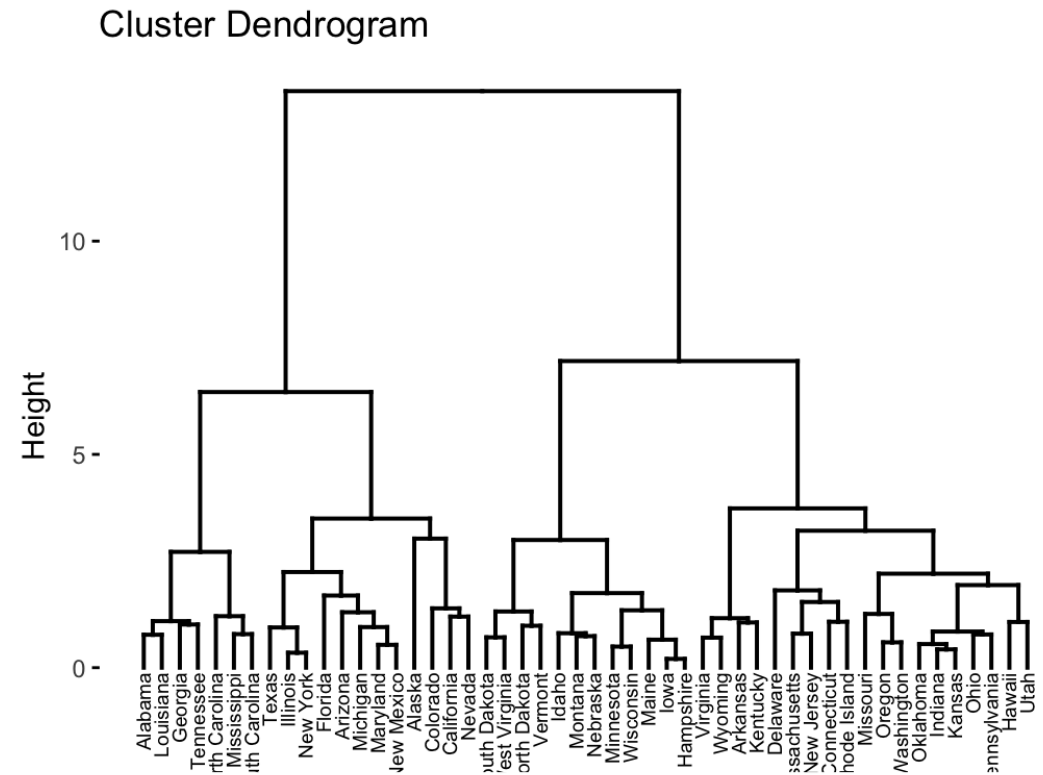


The ML element : EM

- Expectation Maximization algorithm
- Generative model :
 - Clusters represented as mix of Gaussians (means, covariance matrix)
 - Probability for each Gaussian
 - All of which are learned
- Alternating :
 - For each point, calculate probability for belonging to different Gaussians
 - Assign to most likely one
 - Update Gaussians' parameters
 - Repeat

The ML element : HAC

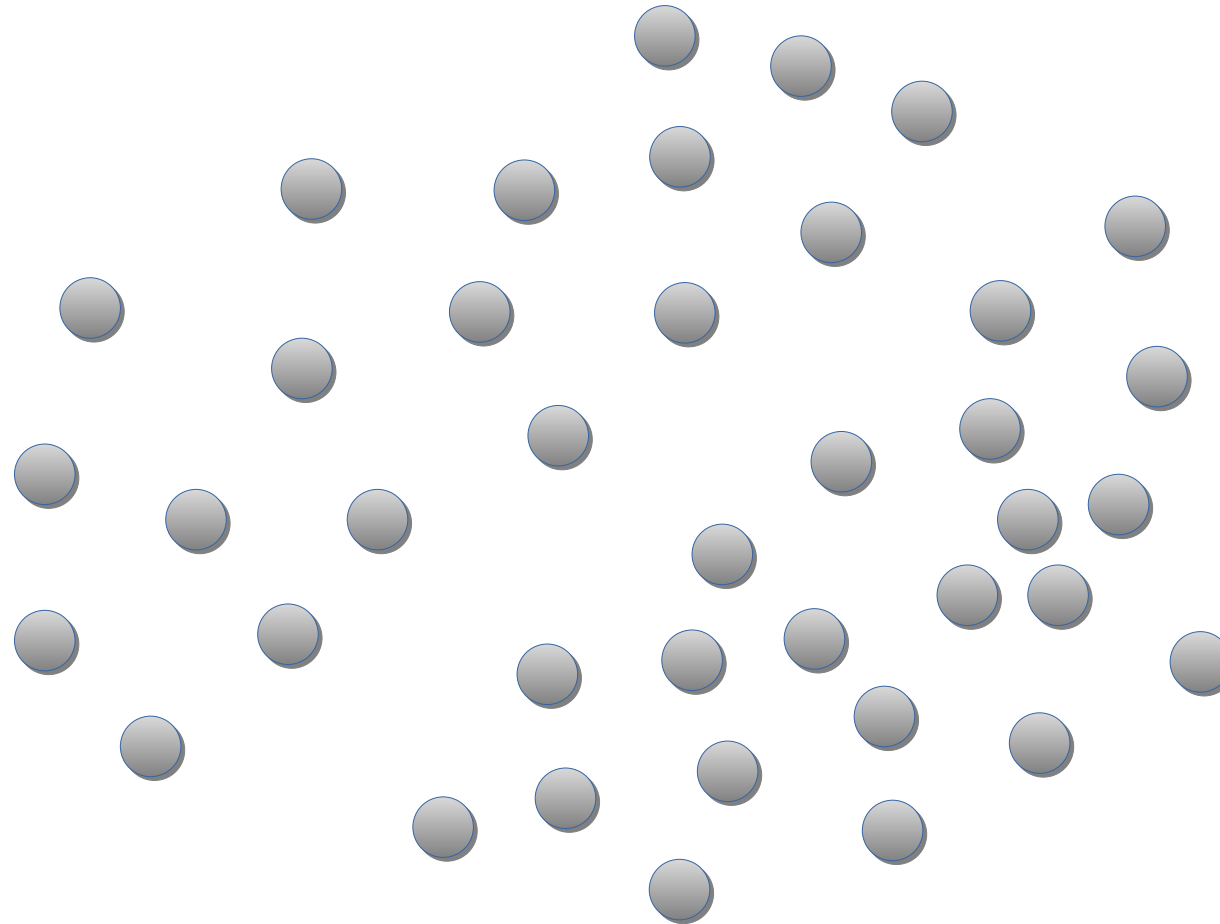
- Hierarchical agglomerative clustering
- In the beginning, each data point its own cluster
- Merge the two closest ones
- Until only a single cluster left
- The « tree » structure is learned



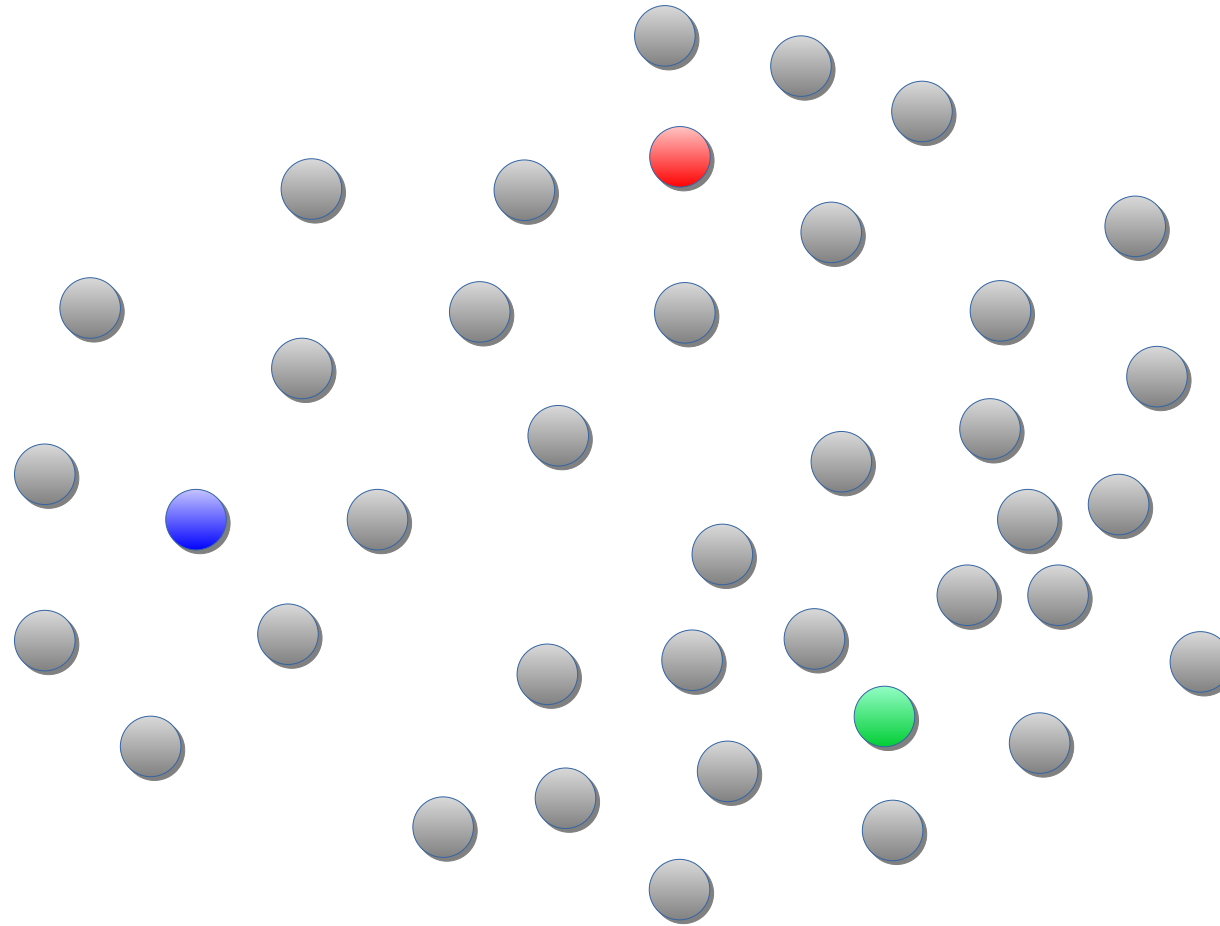
The ML element : k-means clustering

- Cluster represented as « centroids » - mathematical middle of a « cloud » of points
 - Which are learned
- Pick starting centroids at random
- Assign each point to most similar centroid
- Recalculate centroids (average over all cluster points)
- Repeat
- Minimizes sum of intra-cluster distances

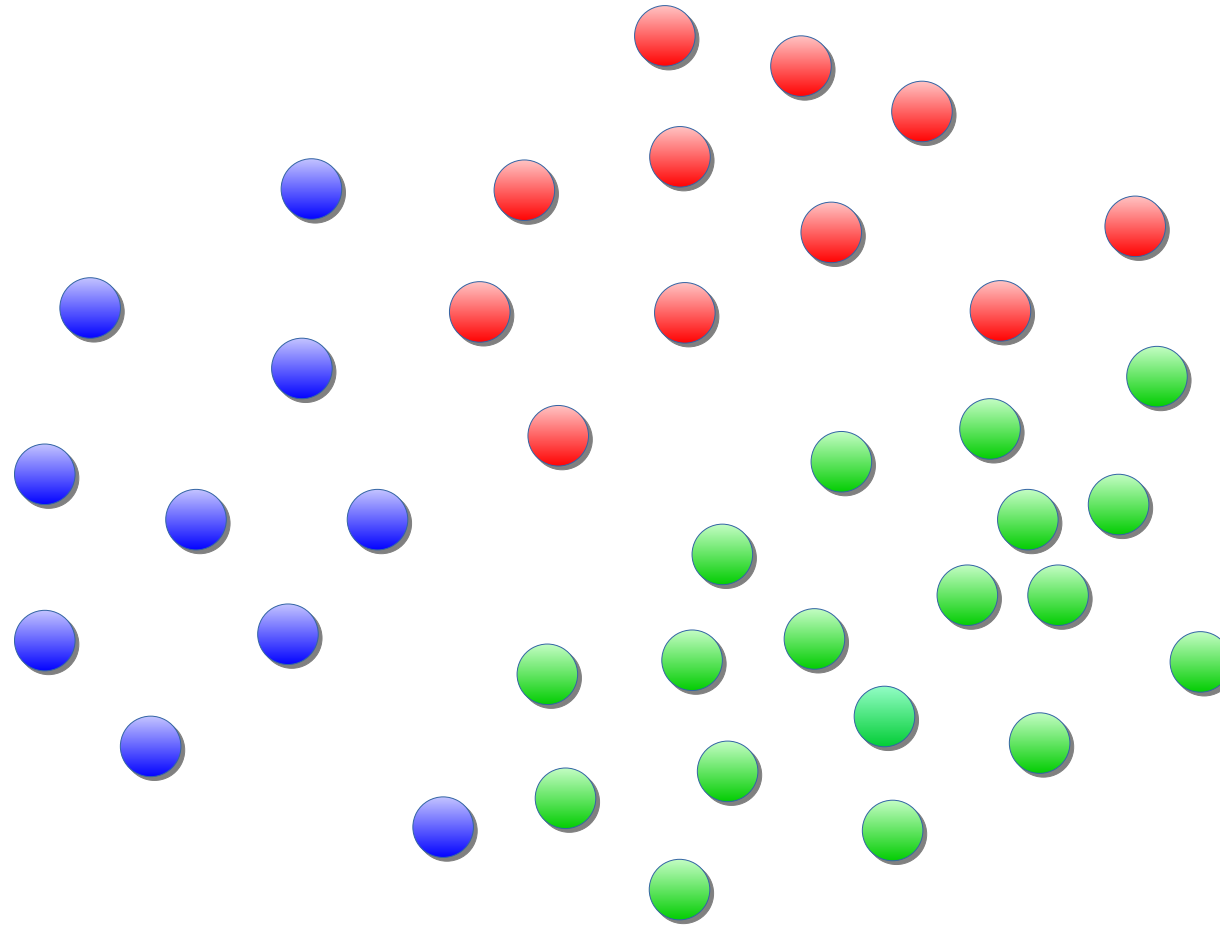
K-Means (1)



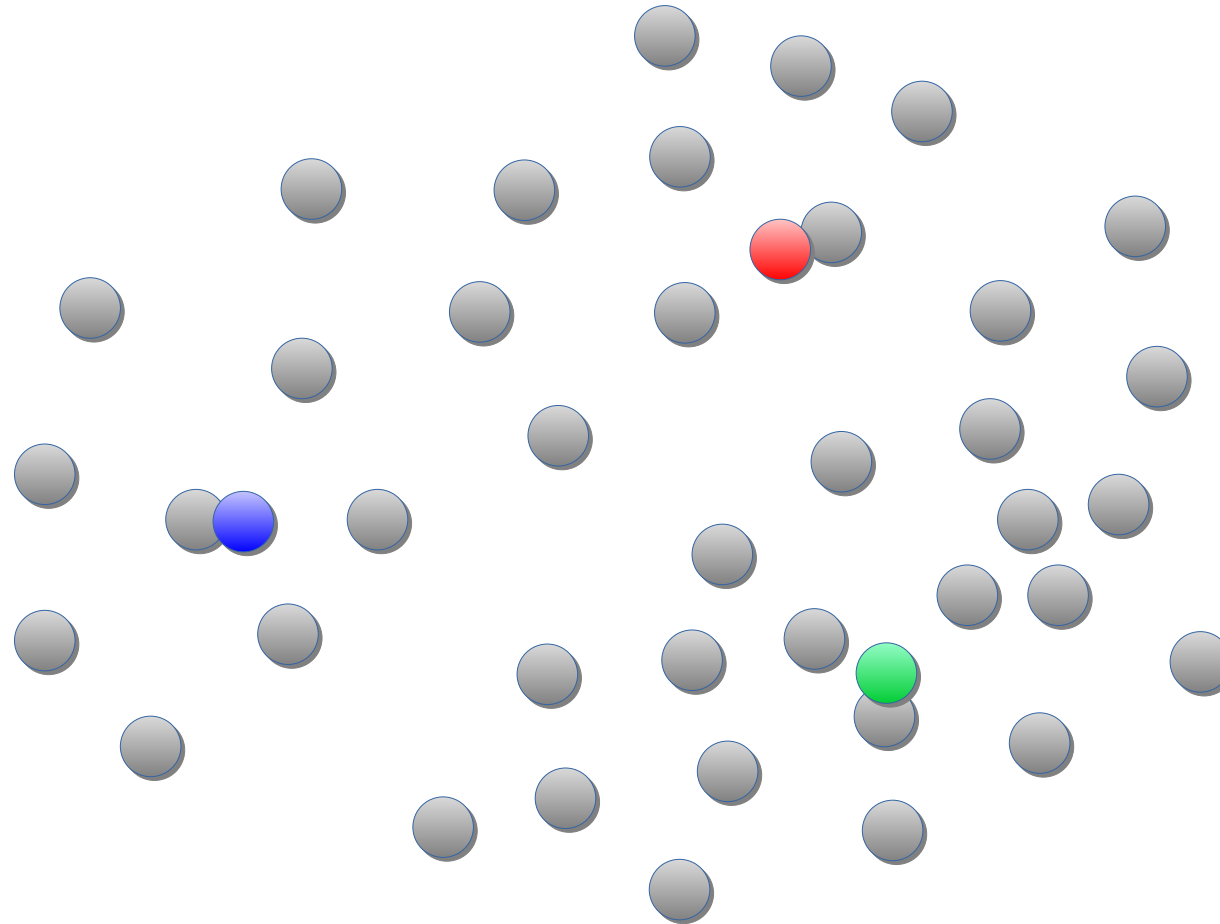
K-Means (2)



K-Means (3)



K-Means (4)



Pass reward

Imbalance problematic

- Shot within the next ten seconds
 - → extreme imbalance between classes : 10489 vs 194664
- Learned via logistic regression

The ML element : logistic regression

- Learned from pass examples : x descriptors, y success/failure (reward)
 - **Supervised** problem
- Logistic regression :
 - Linear regression $y' = \beta + \beta_1 * x_1 + \beta_2 * x_2 + \dots$
 - The β s are learned
 - Passed through logistic (or sigmoid) function $1/(1+e(-y'))$ to get well-calibrated results
- *Result between 0 and 1, interpretable as probabilities*
- Assumes linear relationship

Player assessment

- Individual passes in a play : which player played the one w/highest reward ?
- Passing Plus Minus (PMM) : 1-pass risk if successful, -risk is unsuccessful, normalized per 90 mins
- Difficult pass completion : proportion of passes in 75th percentile completed
- Passes Received Added (PRA) : credit of 1-risk per pass received
- Combine the two as catch-all stat
- Dangerous pass : reward in 75th percentile
 - Again attempted, received
- Evaluated via recourse to « known » players

Actions speak louder than goals : Valuing Player Actions in Soccer (SciSports paper)

- Event stream data
- Game states labeled w/chance of scoring/conceding
 - In « near » future
- Performing actions → moving from one game state to the next
- VAEP : change probability to score + -1*change probability to concede
- Two classifiers to predict $\text{prob}_{\text{score}}$, $\text{prob}_{\text{concedes}}$
 - Learned via form of **gradient boosted trees**

Data for learning

- Label : 1 if scored/conceded within the k next states ($k=10$ in the paper)
- Features : concatenation of a_{i-2}, a_{i-1}, a_i
 - Type, result, body part
 - Start and end locations, distance/angle to goal, distance covered
 - time elapsed since start of game
 - Distance, elapsed time between consecutive actions
 - Game context : goals scored by attacking/defending team, goal difference

Data transformation/
augmentation

The ML element : decision trees

Weather	Temperature	Humidity	Windy	Play tennis
sunny	hot	high	false	No
sunny	hot	high	true	No
cloudy	hot	high	false	Yes
pluvieux	mild	high	false	Yes
rainy	cold	normal	false	Yes
rainy	cold	normal	true	No
cloudy	cold	normal	true	Yes
sunny	mild	high	false	No
sunny	cold	normal	false	Yes
rainy	mild	normal	false	Yes
sunny	mild	normal	true	Yes
cloudy	mild	high	true	Yes
cloudy	hot	normal	false	Yes
rainy	mild	high	true	No

The ML element : decision trees

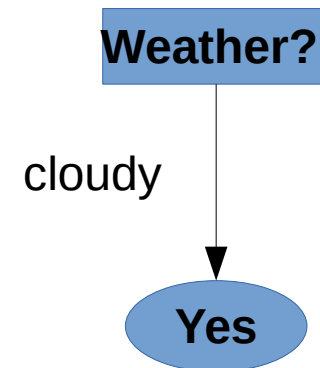
Weather	Temperature	Humidity	Windy	Play tennis
sunny	hot	high	false	No
sunny	hot	high	true	No
cloudy	hot	high	false	Yes
pluvieux	mild	high	false	Yes
rainy	cold	normal	false	Yes
rainy	cold	normal	true	No
cloudy	cold	normal	true	Yes
sunny	mild	high	false	No
sunny	cold	normal	false	Yes
rainy	mild	normal	false	Yes
sunny	mild	normal	true	Yes
cloudy	mild	high	true	Yes
cloudy	hot	normal	false	Yes
rainy	mild	high	true	No

Asks questions, partitions data according to answer

The ML element : decision trees

Weather	Temperature	Humidity	Windy	Play tennis
sunny	hot	high	false	No
sunny	hot	high	true	No
cloudy	hot	high	false	Yes
pluvieux	mild	high	false	Yes
rainy	cold	normal	false	Yes
rainy	cold	normal	true	No
cloudy	cold	normal	true	Yes
sunny	mild	high	false	No
sunny	cold	normal	false	Yes
rainy	mild	normal	false	Yes
sunny	mild	normal	true	Yes
cloudy	mild	high	true	Yes
cloudy	hot	normal	false	Yes
rainy	mild	high	true	No

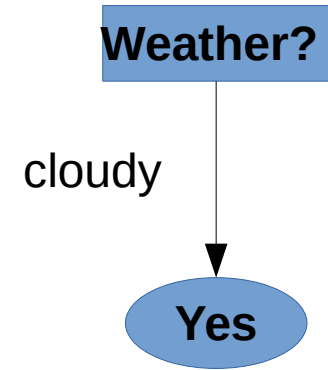
Asks questions, partitions data according to answer



The ML element : decision trees

Weather	Temperature	Humidity	Windy	Play tennis
sunny	hot	high	false	No
sunny	hot	high	true	No
cloudy	hot	high	false	Yes
pluvieux	mild	high	false	Yes
rainy	cold	normal	false	Yes
rainy	cold	normal	true	No
cloudy	cold	normal	true	Yes
sunny	mild	high	false	No
sunny	cold	normal	false	Yes
rainy	mild	normal	false	Yes
sunny	mild	normal	true	Yes
cloudy	mild	high	true	Yes
cloudy	hot	normal	false	Yes
rainy	mild	high	true	No

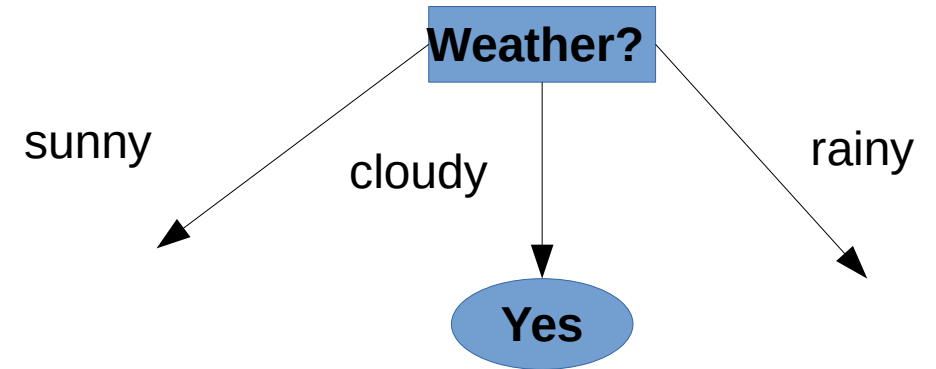
Asks questions, partitions data according to answer



The ML element : decision trees

Weather	Temperature	Humidity	Windy	Play tennis
sunny	hot	high	false	No
sunny	hot	high	true	No
cloudy	hot	high	false	Yes
pluvieux	mild	high	false	Yes
rainy	cold	normal	false	Yes
rainy	cold	normal	true	No
cloudy	cold	normal	true	Yes
sunny	mild	high	false	No
sunny	cold	normal	false	Yes
rainy	mild	normal	false	Yes
sunny	mild	normal	true	Yes
cloudy	mild	high	true	Yes
cloudy	hot	normal	false	Yes
rainy	mild	high	true	No

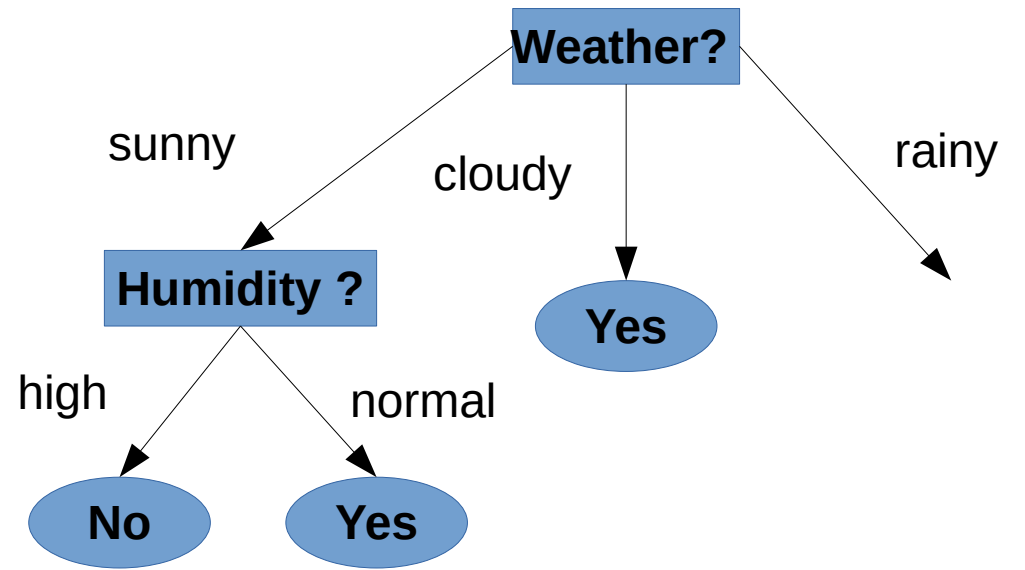
Asks questions, partitions data according to answer



The ML element : decision trees

Weather	Temperature	Humidity	Windy	Play tennis
sunny	hot	high	false	No
sunny	hot	high	true	No
cloudy	hot	high	false	Yes
pluvieux	mild	high	false	Yes
rainy	cold	normal	false	Yes
rainy	cold	normal	true	No
cloudy	cold	normal	true	Yes
sunny	mild	high	false	No
sunny	cold	normal	false	Yes
rainy	mild	normal	false	Yes
sunny	mild	normal	true	Yes
cloudy	mild	high	true	Yes
cloudy	hot	normal	false	Yes
rainy	mild	high	true	No

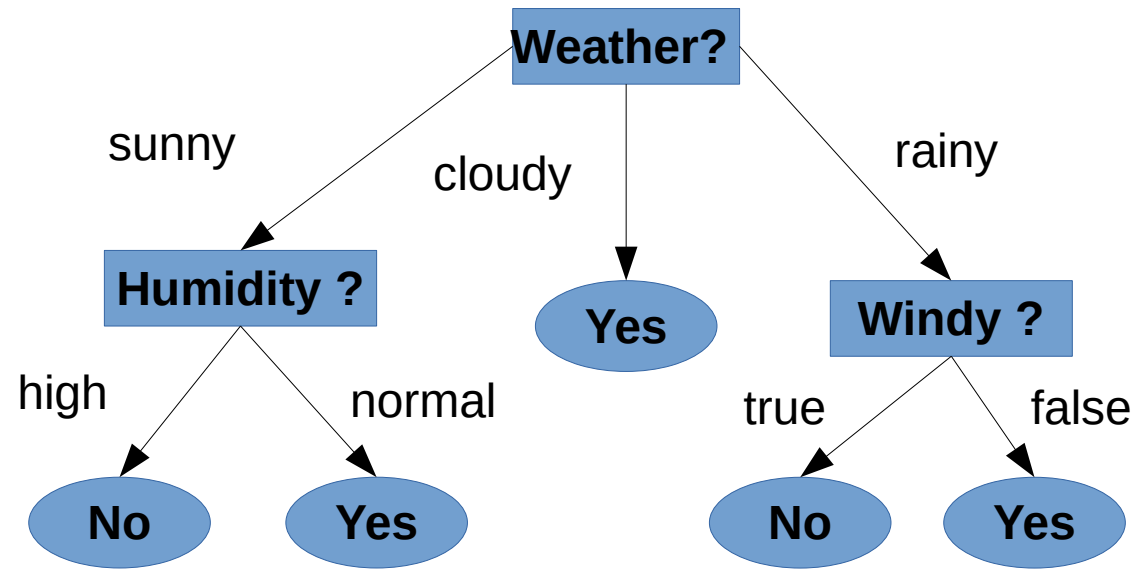
Asks questions, partitions data according to answer



The ML element : decision trees

Weather	Temperature	Humidity	Windy	Play tennis
sunny	hot	high	false	No
sunny	hot	high	true	No
cloudy	hot	high	false	Yes
pluvieux	mild	high	false	Yes
rainy	cold	normal	false	Yes
rainy	cold	normal	true	No
cloudy	cold	normal	true	Yes
sunny	mild	high	false	No
sunny	cold	normal	false	Yes
rainy	mild	normal	false	Yes
sunny	mild	normal	true	Yes
cloudy	mild	high	true	Yes
cloudy	hot	normal	false	Yes
rainy	mild	high	true	No

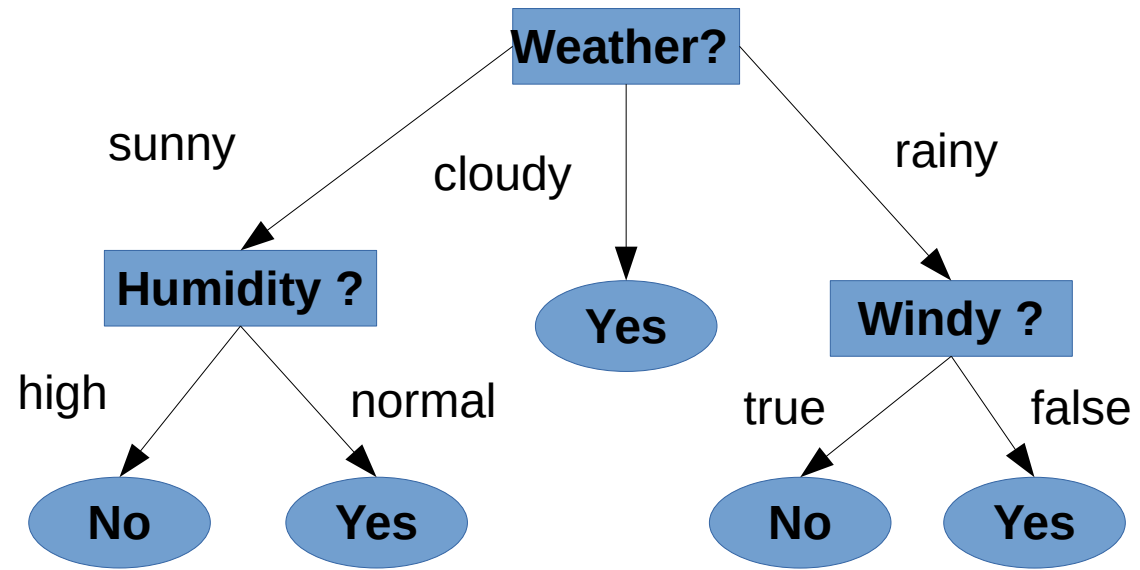
Asks questions, partitions data according to answer



The ML element : decision trees

Weather	Temperature	Humidity	Windy	Play tennis
sunny	hot	high	false	No
sunny	hot	high	true	No
cloudy	hot	high	false	Yes
pluvieux	mild	high	false	Yes
rainy	cold	normal	false	Yes
rainy	cold	normal	true	No
cloudy	cold	normal	true	Yes
sunny	mild	high	false	No
sunny	cold	normal	false	Yes
rainy	mild	normal	false	Yes
sunny	mild	normal	true	Yes
cloudy	mild	high	true	Yes
cloudy	hot	normal	false	Yes
rainy	mild	high	true	No

Asks questions, partitions data according to answer



Tests (questions) learned based on quality function

The ML element : boosting

- Learn a succession of classifiers
- Change weight of (label of) mis-classified data points
- Adaboost :
 - Increase weight, instead of 1 mis-classified, have 1.2 mis-classified
- Gradient boost :
 - Cost function with nice derivative
 - Applied to size of error
 - Change label : residual = actual – predicted values
 - Well-classified goes towards 0

Player rating

- Sum of value of all actions performed by player during time frame, normalized for 90 mins
 - Can be limited to particular action type
- Compared to goals + assists, salaries
- Replacement assessment, Neymar, Cristiano Ronaldo

Choke or shine (SciSports paper)

Data transformation/
augmentation

- Who performs well (or not) under pressure ?
- Pre-game pressure :
 - Labels : 19 soccer experts rank pairs out of 170 random games (inter-expert agreement 79.79%)
 - Team ambition : **k-means** clustering into 4 groups based on previous result, transfer value of top-20 players, spending on loans, Football manager reputation score
 - Game importance : Elo-based Monte Carlo season simulation, Kendall-Tau correlation between game and season outcome
 - Recent performance : number of points obtained, deviation from expected performance based on Elo
 - Context : location, rivalry (Football manager), attendance, since when coach is there
- Learned used **Gradient Boosted Trees (LambdaMART)** :
73.91 % agreement w/experts

In-game pressure

- Pressure increases if goal improves chance for favorable outcome, decreases when goal has small influence
- In-game pressure $\text{winProb} + \text{tieProb}$ **if** team scored in this moment
- Win probability model
 - Future goals based on current state
 - Number of goals scored, goal difference, # yellow/red cards, difference in Elo rating, avg # attacking passed/avg # won duels in previous 10 frames (1 match = 100 frames)
- Learned via **Automatic Differentiation Variational Inference**

Measuring player performance

- Contribution : improve chance to win ?
 - VAEP
- Quality : were there better options ?
 - Expected contribution over all possible actions – actual contribution
 - Needs probability of success
 - Successful/failed actions, described by locations, body part, locations for previous actions
 - **GBT** as learner
- Execution : how well done ?
 - Actual outcome – predicted probability

Player assessment

- Pressure levels :
 - High : top-20 %
 - Normal : middle-60 %
 - Low : bottom-20 %
- Contribution rating
 - Avg/90 mins
 - Avg/90 per action type
 - Percentile rank/90
- Decision rating
 - Avg
 - Avg per type
- Execution rating
 - Pct rank per type

Some results

- Riyadh Mahrez better in high-pressure
 - Rachid Ghezzal not a good replacement
- Tom Dwyer commits useless fouls under pressure
- Neymar's performance declines under pressure
 - Worse decisions

Deephoops : Evaluating Micro-Actions in Basketball Using Deep Feature Representations of Spatio-Temporal Data

- Micro-actions : screen, pass, backdoor-cut etc
- Player tracking data
 - → moments : vector w/positions of all players, x, y, z of ball, seconds on shot-clock, indicator for the 5 offensive/defensive players, polar coordinates w.r.t. basket of defending team
 - Window of size $T=128$ (typically several actions during a single possession), annotated with terminal action (field goal attempt, shooting foul, No-shooting foul, turnover, null)

Data transformation

- Distribution of terminal actions learned via **Long Short-Term Memory** neural networks

- Supervised problem

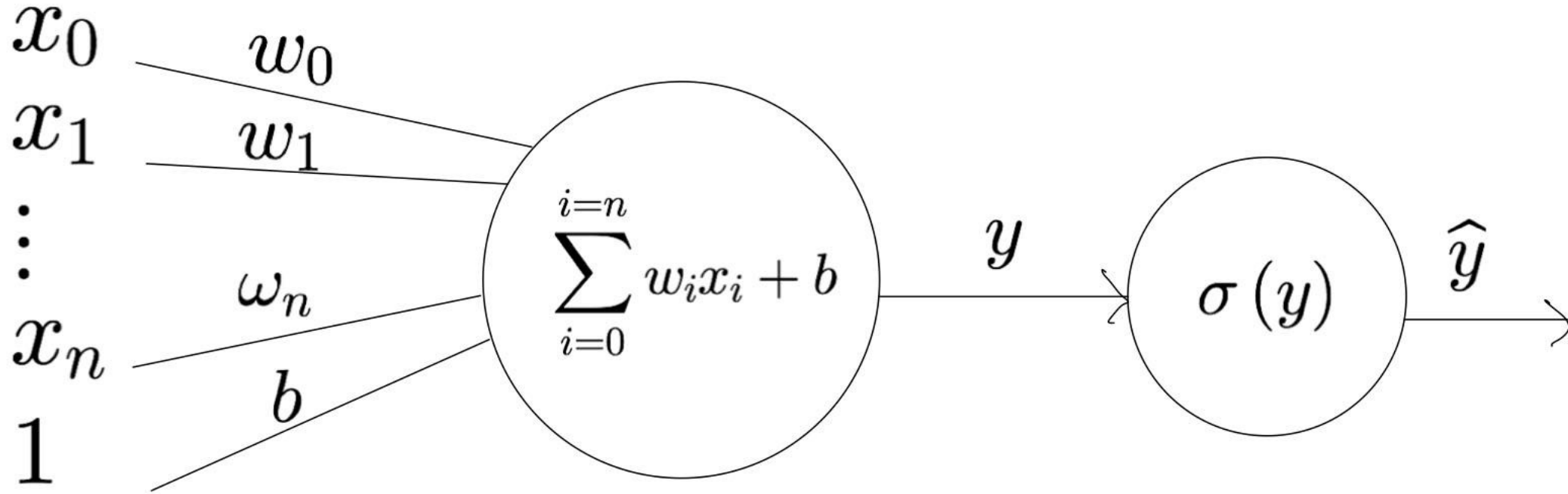
32 cells, 3 layers

- Map terminal actions to expected points :

- Baseline : points/possessions
- Difference → value of terminal action

« Null » outweighs rest 600 : 1
→ under-sampling

The ML element : Neural networks

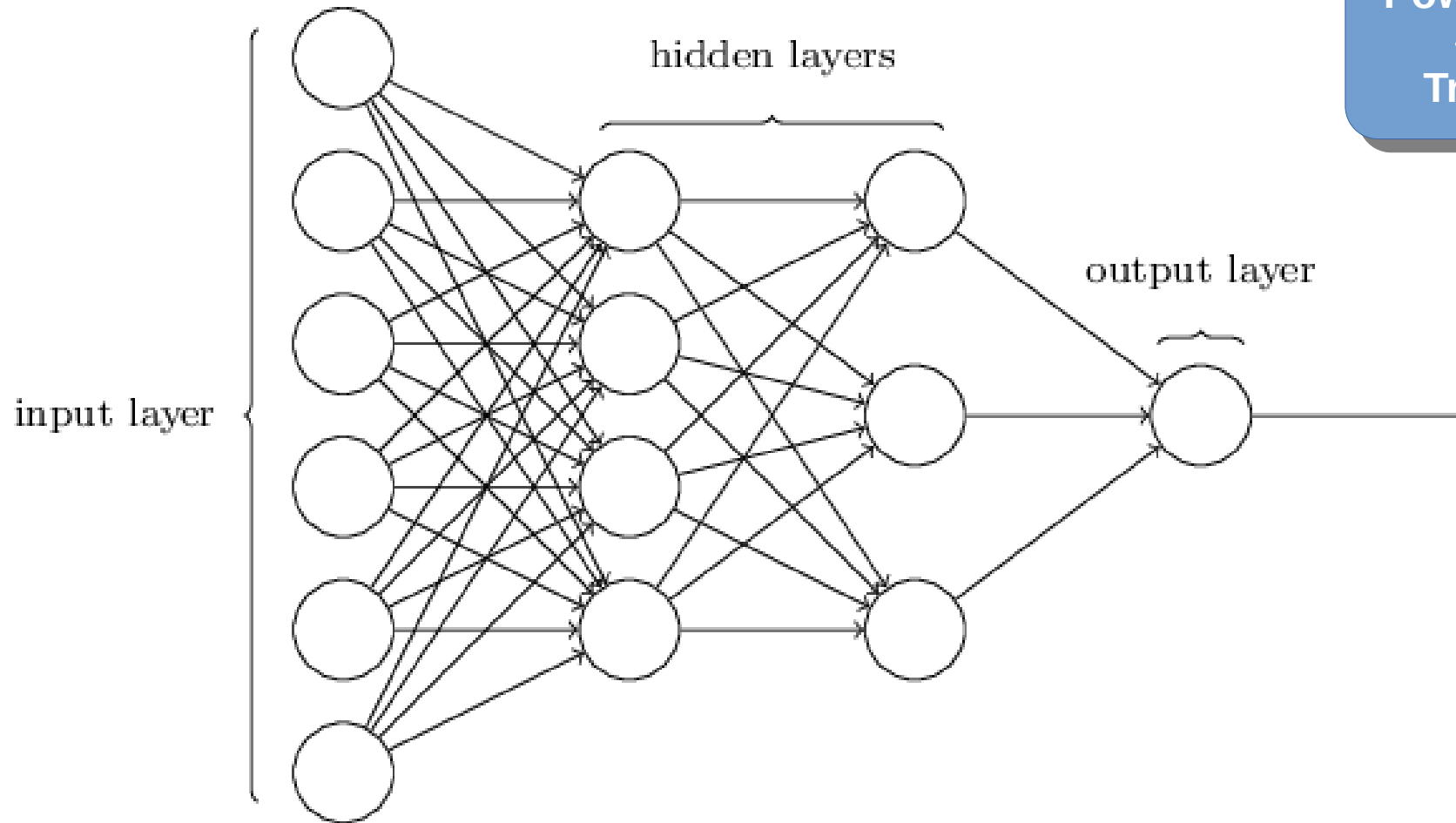


$$y' = \sigma(b + w_0 * x_0 + w_1 * x_1 + \dots w_n * x_n)$$

Linear regression pass through function ~ logistic regression

Rosenblatt '58

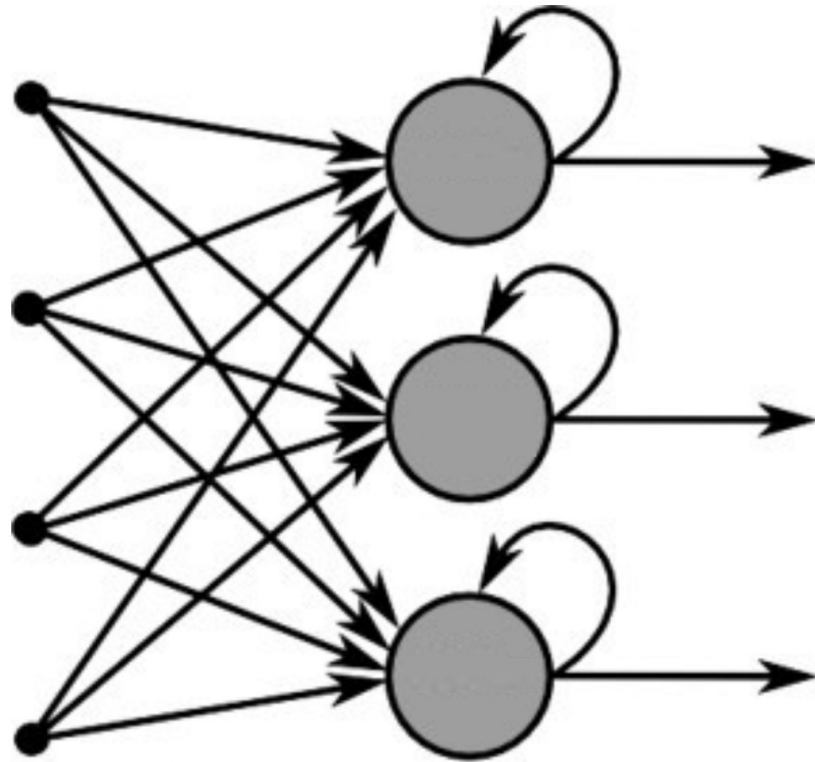
Neural networks (2)



Power comes from combination
« feed-forward » network
Trained by gradient descent

Rumelhart, Hinton & Williams '86

Dealing w/sequential information : Recurrent neural networks

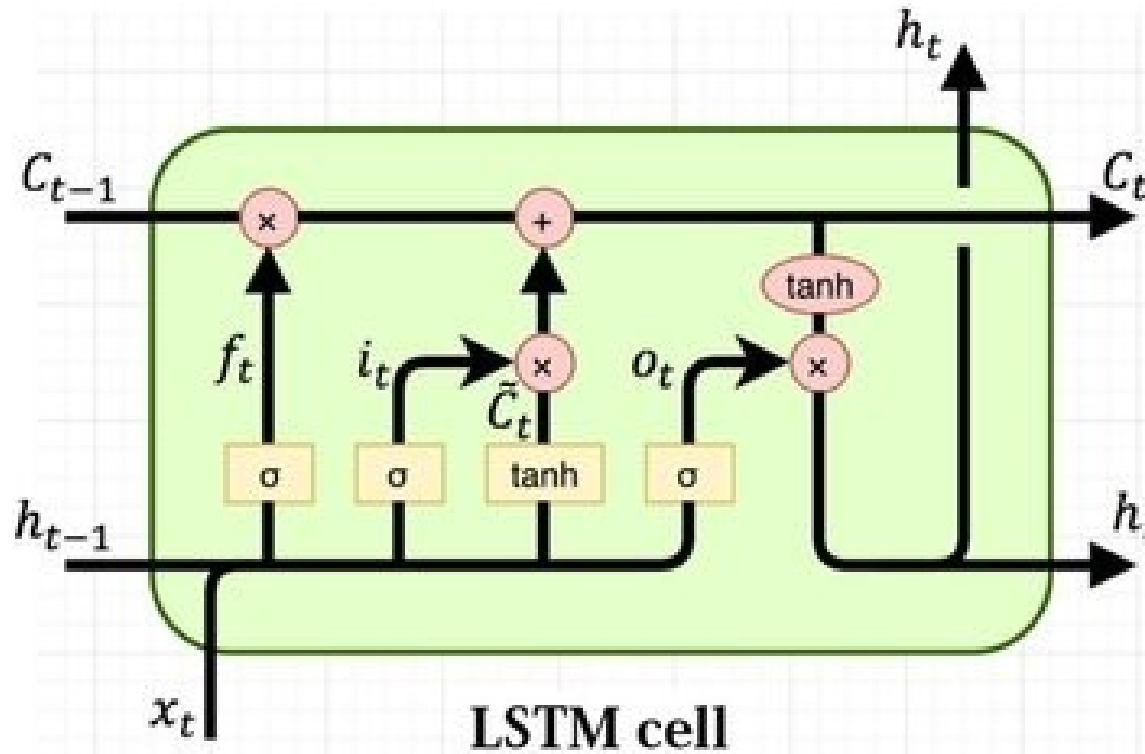


Recurrent Neural Network

Problem during learning : error feedback may become overwhelming or disappear for long time lags (>10 steps)

Older than '89

The ML element : LSTM



$$i_t = \sigma(x_t U^i + h_{t-1} W^i)$$

$$f_t = \sigma(x_t U^f + h_{t-1} W^f)$$

$$o_t = \sigma(x_t U^o + h_{t-1} W^o)$$

$$\tilde{C}_t = \tanh(x_t U^g + h_{t-1} W^g)$$

$$C_t = \sigma(f_t * C_{t-1} + i_t * \tilde{C}_t)$$

$$h_t = \tanh(C_t) * o_t$$

Gated access to cell's « memory » :

- input
- output
- forget

originally for whole blocks of cells

Hochreiter & Schmidhuber '97
Gers et al. '00

Player embedding

- Also neural network
- Learned in parallel
- Players w/similar contributions to distribution terminal actions « close » (similar) in learned space
 - Unsupervised problem
- Concatenated w/LSTM output
- Fed into additional dense neural network layer

Player assessment

- Did player action increase/decrease expected points ?

Bhostgusters : Realtime Interactive Play Sketching with Synthesized NBA Defenses (Disney research + Stats Inc)

- Where can we expect a defender to be ?
- Player-tracking data
 - Possessions of ten 2D trajectories, one 3D trajectory
 - Game clock, shot clock, player fouls, seconds logged per player
- Modeled by 5 2-layer LSTMs
 - Label to be learned is next position of player
 - Pre-training : predict where player is at next time step, given all other players' positions
 - Individual training : predict positions several steps into future, given all others'
 - Joint training : like individual but w/predictions of other LSTMs as input

ANN can have more than one output node...

Labels : Expected points to assess actions

- (Potential) shot location
- Ball state
- Distance of two nearest defenders
- Learned via (parsimonious since only few descriptors) regression

Player assessment

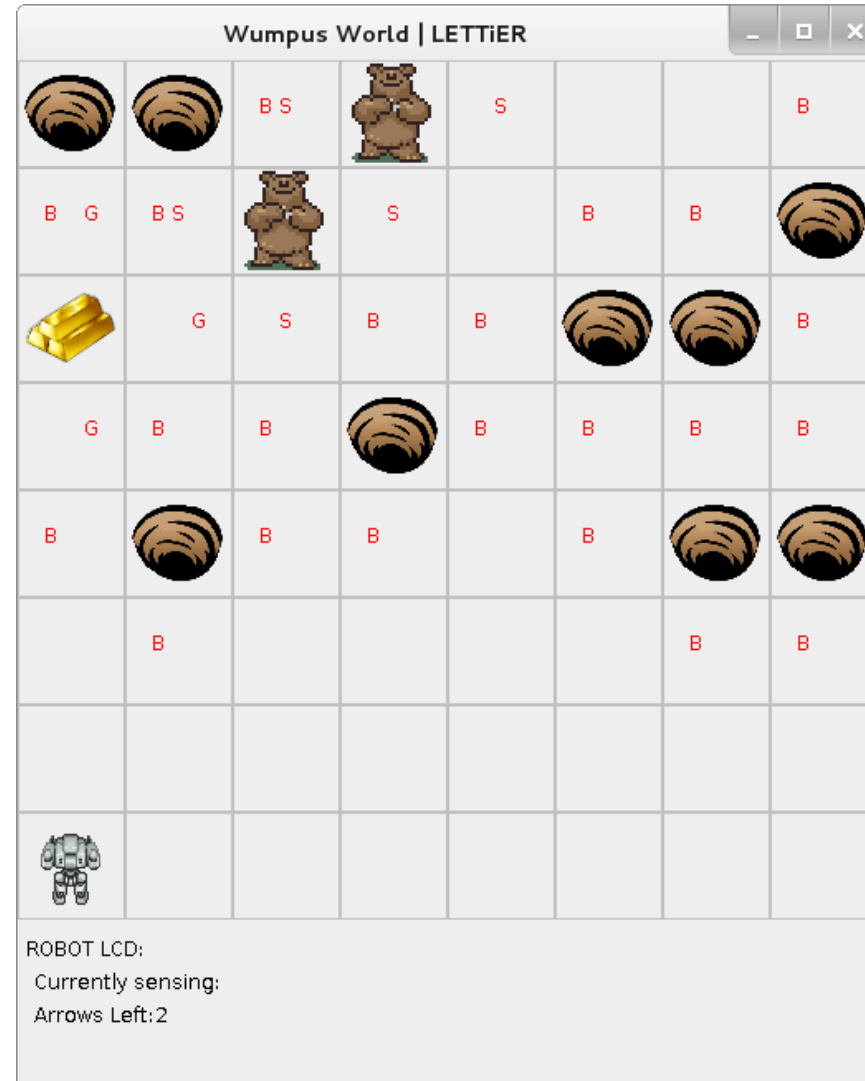
- Was a defending player where he was expected to be ?
- Did a defending player's positioning best reduce EPM ?
- What offensive actions lead to best EPM ?

Deep Reinforcement Learning in Ice Hockey for Context-Aware Player Evaluation

- Event data
- Reinforcement learning
 - 3-dim reward vector : goal scored by home/away/neither
 - 13 possible actions
- Goal-scoring episodes
 - From beginning of game/after goal has been scored
 - To goal scored/end of game
- Learned by DNN
 - Three output nodes
 - 3 hidden layers, 1 LSTM, 2 fully connected
- Temporal difference learning

The ML element : reinforcement learning

- Neither supervised nor unsupervised
 - Label « changes » with distance from goal
- States of the world
 - Actions → state transitions
 - Action-value function $Q(\text{state}, \text{action})$



Q-learning : '89

Player assessment

- Goal Impact Metric (GIM)
 - Change in goal scoring probability of players' team from s_{t-1} to s_t
 - Summed over match or season, normalized
- Evaluated via
 - Comparison to NHL stars
 - Salaries
 - Currently used statistics (correlation)

Takeaway message

- Getting the right data is important
 - And hard/time-consuming/expensive
- Preparing the data might be even more so
 - Requires understanding of the domain
- Deep learning offers powerful methods
 - But it's not only set of tools
- Evaluation methods are purely academical so far