Computazione per l'interazione naturale: Modelli dinamici

Corso di Interazione uomo-macchina II

Prof. Giuseppe Boccignone

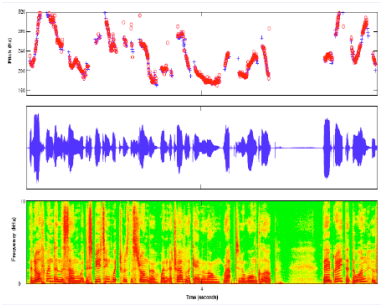
Dipartimento di Scienze dell'Informazione
Università di Milano

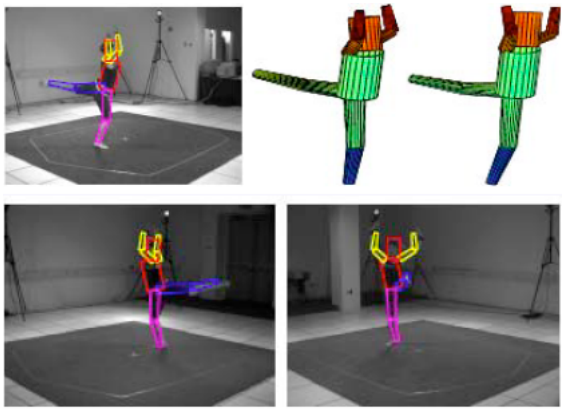
boccignone@di.unimi.it
http://boccignone.di.unimi.it/IUM2_2014.html

Modelli dinamici

	Supervised Learning	*Unsupervised Learning*
Discrete	classification or categorization Discrete Output y $\in \{1, \ldots, K\}$	clustering
Continuous	regression Continuous Output $y \in R$	dimensionality reduction

Modelli dinamici





Processi temporali

- N stati o categorie

$$x_t \in \{1, 2, \ldots, N\} \longrightarrow$$ stato al tempo t

- Distribuzione congiunta fattorizzabile come

$$p(x_0, x_1, \ldots, x_T) = p(x_0) \prod_{t=1}^{T} p(x_t \mid x_0, \ldots, x_{t-1})$$

Processi di Markov

- Il prossimo stato dipende dallo stato presente

$$p(x_{t+1} \mid x_0, \ldots, x_t) = p(x_{t+1} \mid x_t)$$

- Condizionatamente al presente, passato e futuro sono indipendenti

$$\begin{aligned} & p(x_0, \ldots, x_{t-1}, x_{t+1}, \ldots, x_T \mid x_t) \\ = \; & p(x_0, \ldots, x_{t-1} \mid x_t) p(x_{t+1}, \ldots, x_T \mid x_t) \end{aligned}$$

Matrici di transizione di stato

- Catena di Markov stazionaria con N stati descritta da una matrice di transizione

$$Q = \begin{bmatrix} q_{11} & q_{12} & q_{13} \\ q_{21} & q_{22} & q_{23} \\ q_{31} & q_{32} & q_{33} \end{bmatrix}$$

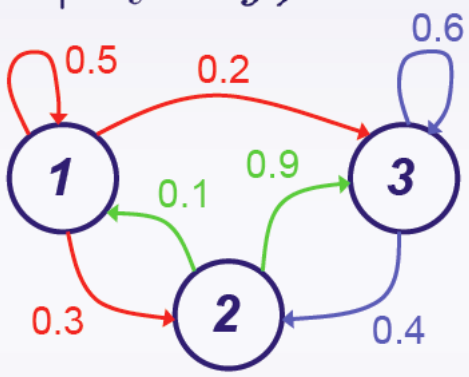
$$q_{ij} \triangleq p(x_{t+1} = i \mid x_t = j)$$

$$q_{ij} \geq 0 \qquad \sum_{i=1}^{N} q_{ij} = 1$$

Diagrammi di transizione fra stati

$$q_{ij} \triangleq p(x_{t+1} = i \mid x_t = j)$$

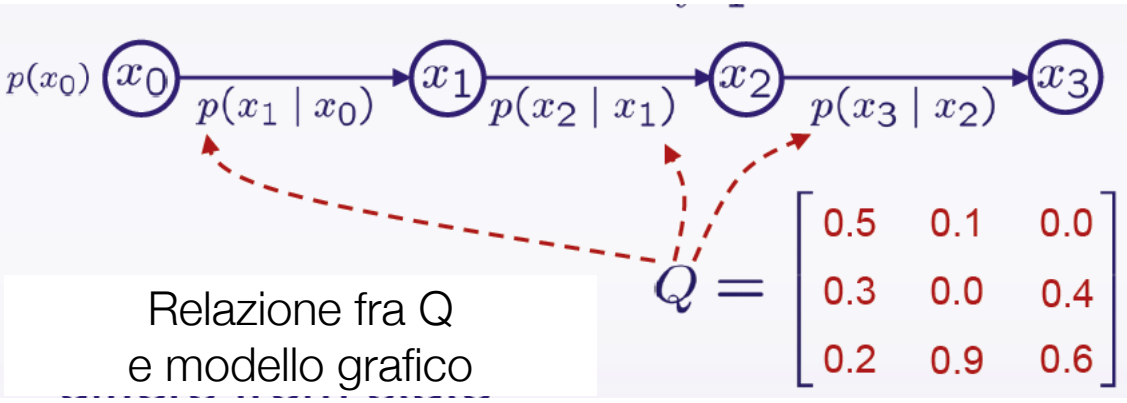
$$Q = \begin{bmatrix} 0.5 & 0.1 & 0.0 \\ 0.3 & 0.0 & 0.4 \\ 0.2 & 0.9 & 0.6 \end{bmatrix}$$



Modello grafico di una catena di Markov

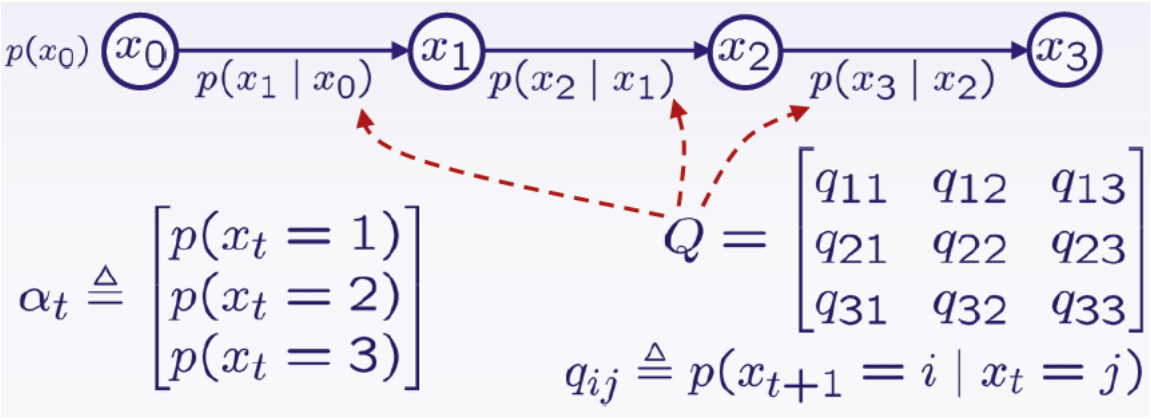
- Da non confondere con il diagramma di transizione fra stati

$$p(x_0, x_1, \ldots, x_T) = p(x_0) \prod_{t=1}^{T} p(x_t \mid x_{t-1})$$



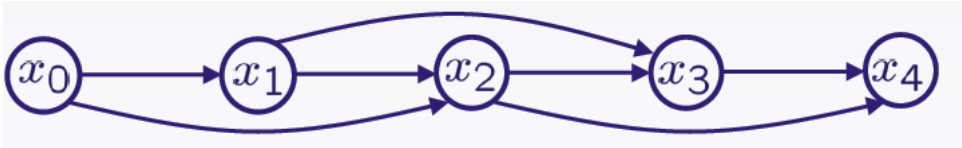
Relazione fra Q
e modello grafico

Statistiche delle catene di Markov



$$\alpha_1(i) = \sum_{j=1}^{N} q_{ij}\alpha_0(j) \qquad \begin{cases} \alpha_1 = Q\alpha_0 \\ \alpha_t = Q^t\alpha_0 \\ \alpha_\infty = ??? \end{cases}$$

Catene di ordine superiore al primo

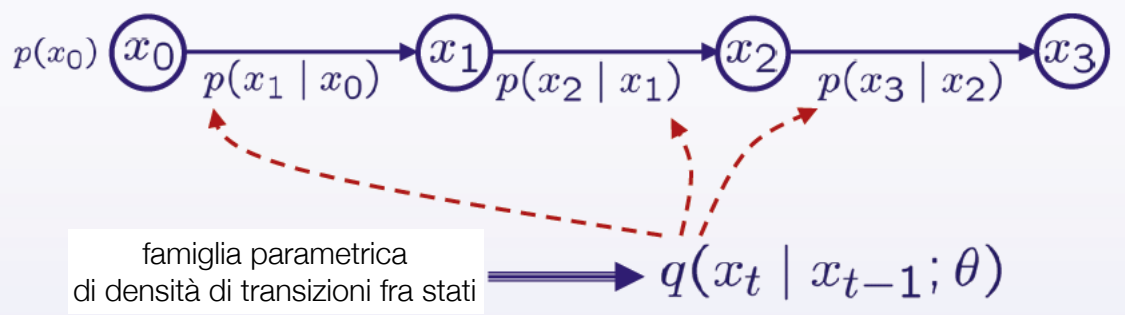


$$p(x_0, x_1, \ldots, x_T) = p(x_0) \prod_{t=1}^{T} p(x_t \mid x_{t-1}, x_{t-2})$$

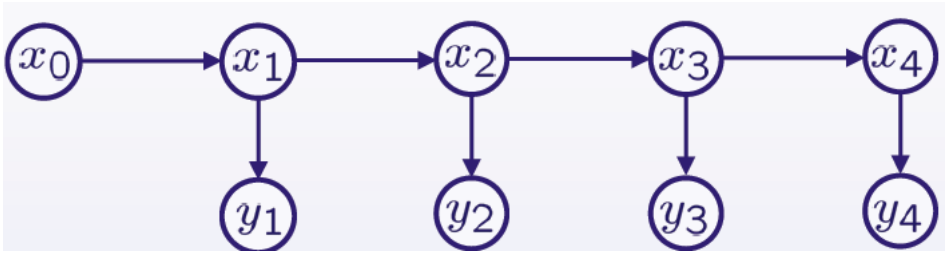
$$\bar{x}_t \triangleq \{x_t, x_{t-1}\} \qquad p(\bar{x}) = p(\bar{x}_1) \prod_{t=2}^{T} p(\bar{x}_t \mid \bar{x}_{t-1})$$

Processi dinamici a stati continui

- Gli stati sono definiti in uno spazio euclideo continuo: $x_t \in \mathbb{R}^d$



Modelli nascosti di Markov (Hidden Markov Models, HMM)

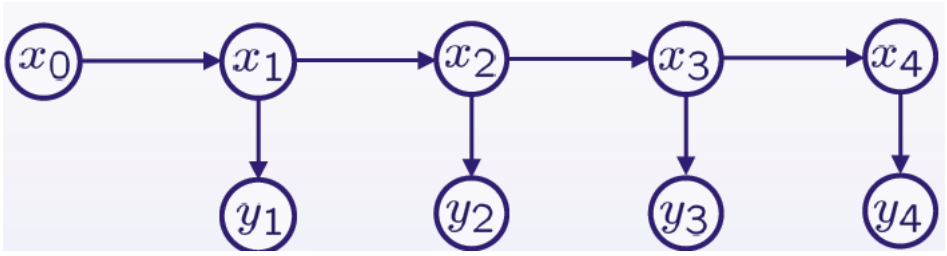


Stati nascosti

Processi osservati

$$p(x_0, x_1, \ldots, x_T) = p(x_0) \prod_{t=1}^{T} p(x_t \mid x_{t-1}) p(y_t \mid x_t)$$

Modelli nascosti di Markov (Hidden Markov Models, HMM)



Stati nascosti

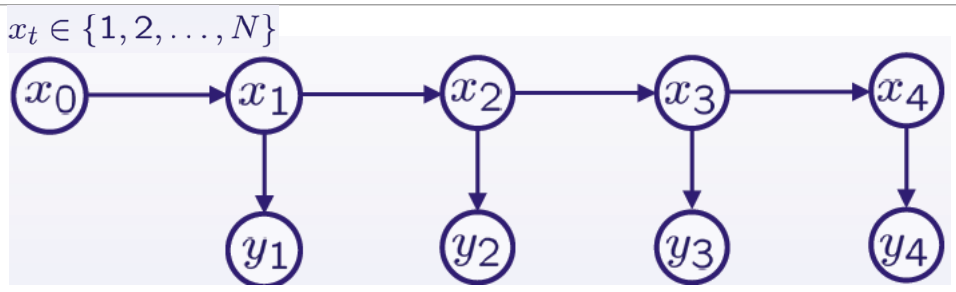
Processi osservati

$$p(x_0, x_1, \ldots, x_T) = p(x_0) \prod_{t=1}^{T} p(x_t \mid x_{t-1}) p(y_t \mid x_t)$$

- Dato x_t le osservazioni passate non ci dicono nulla di più sul futuro

$$p(y_t, y_{t+1}, \ldots \mid x_t, y_{t-1}, y_{t-2}, \ldots) = p(y_t, y_{t+1}, \ldots \mid x_t)$$

Modelli nascosti di Markov //stati discreti



Stati nascosti

Processi osservati

$$p(x_0, x_1, \ldots, x_T) = p(x_0) \prod_{t=1}^{T} p(x_t \mid x_{t-1}) p(y_t \mid x_t)$$

- Si associa a ciascuno degli N stati una differente probabilità di osservazione (emissione)

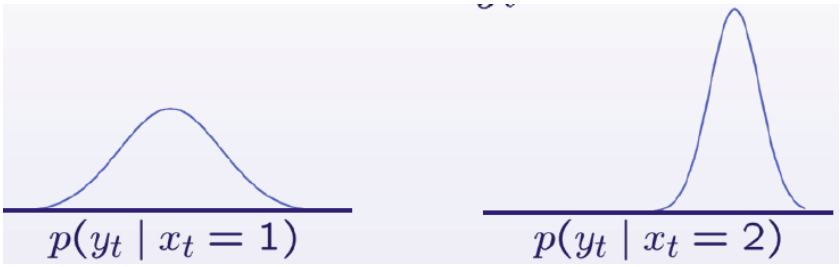
$$p(y_t \mid x_t = 1) \qquad p(y_t \mid x_t = 2) \quad \cdots$$

Modelli nascosti di Markov //osservazioni: discrete e continue

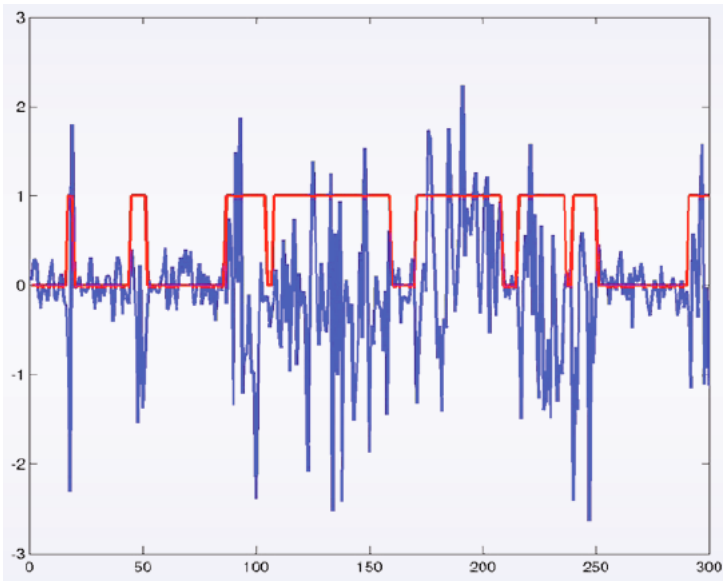
- Osservazioni discrete: $y_t \in \{1, 2, \ldots, M\}$

$$p(y_t \mid x_t = 1) = \begin{bmatrix} 0.3 \\ 0.1 \\ 0.5 \\ 0.1 \end{bmatrix} \qquad p(y_t \mid x_t = 2) = \begin{bmatrix} 0.2 \\ 0.2 \\ 0.1 \\ 0.5 \end{bmatrix}$$

- Osservazioni continue: $y_t \in \mathbb{R}^k$



Esempio



$$x_t \in \{0, 1\}$$

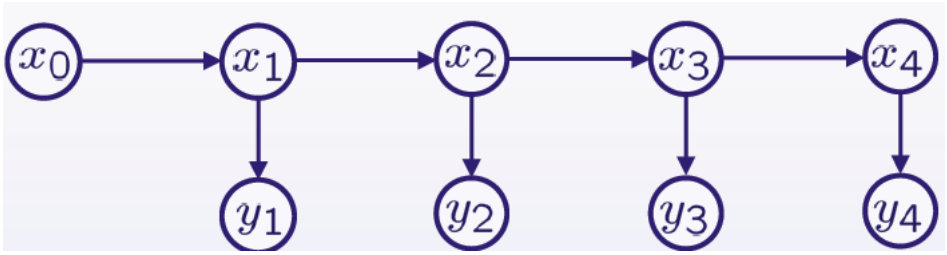
$$Q = \begin{bmatrix} 1-\epsilon & \epsilon \\ \epsilon & 1-\epsilon \end{bmatrix}$$

$$y_t \sim \mathcal{N}(0, \sigma^2_{x_t})$$

$\sigma^2_0 =$ *small*

$\sigma^2_1 =$ *large*

Modelli nascosti di Markov //Inferenza



Stati nascosti

Processi osservati

- Cosa possiamo inferire sugli stati da una sequenza osservata di dati?
  - Filtraggio (analisi on line)

$$p(x_t \mid y_1, y_2, \ldots, y_t) \qquad t = 1, 2, \ldots$$

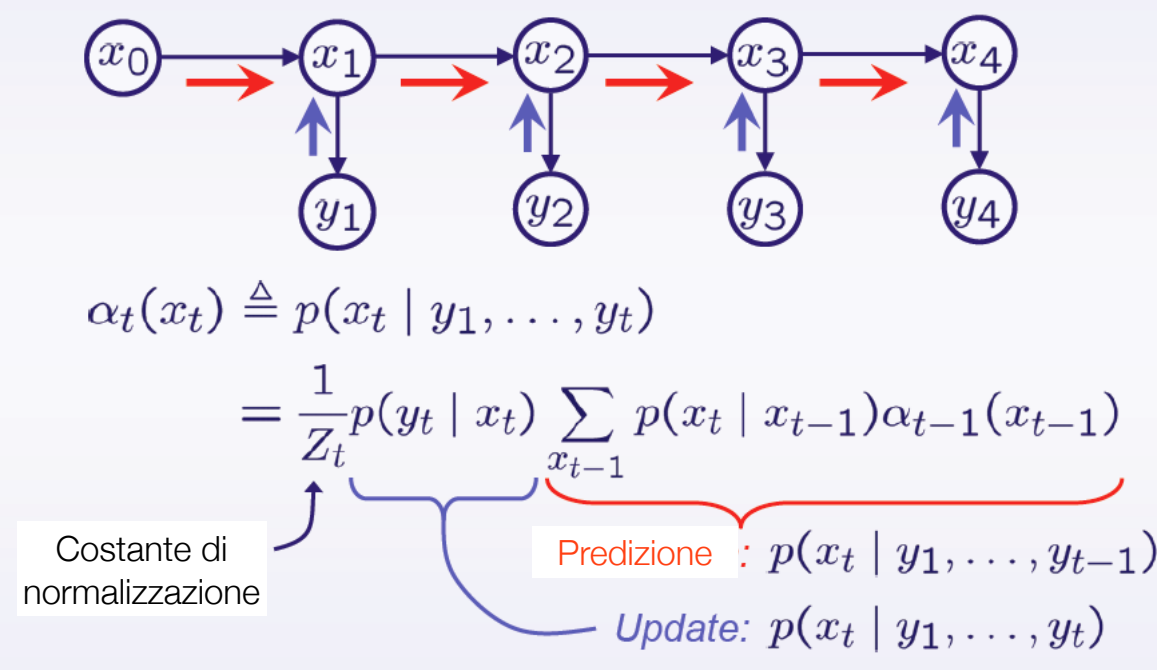
  - Smoothing (analisi batch)

$$p(x_t \mid y_1, y_2, \ldots, y_T) \qquad t = 1, 2, \ldots, T$$

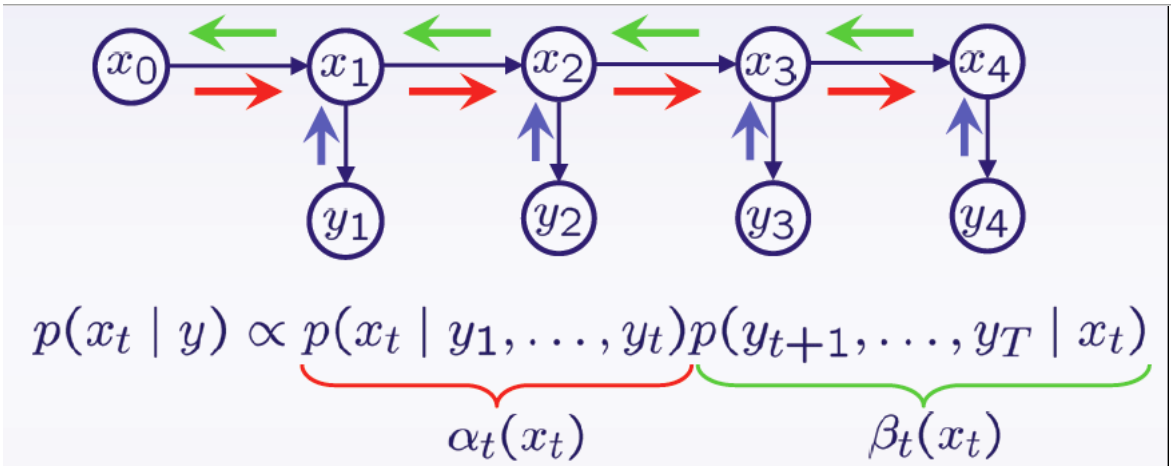
Problemi di inferenza risolvibili con HMM

- Filtraggio $P(X_t|y_{1:t}) \propto P(y_t|X_t) \left[\sum_{x_{t-1}} P(X_t|x_{t-1}) P(x_{1:t-1}|y_{1:t-1})\right]$
- Smoothing $P(X_{t-l}|y_{1:t})$
- Decodifica $x^*_{1:t} = \underset{x_{1:t}}{\operatorname{argmax}}\, P(x_{1:t}|y_{1:t})$
- Classificazione $C^*(y_{1:T}) = \underset{C}{\operatorname{argmax}}\, P(y_{1:T}|C) P(C)$

HMM discreti: //filtraggio



HMM discreti: //smoothing



- L'algoritmo forward-backward aggiorna il filtraggio con una ricorsione indietro nel tempo

$$\beta_t(x_t) = \frac{1}{Z_t} \sum_{x_{t+1}} p(x_{t+1} \mid x_t) p(y_{t+1} \mid x_{t+1}) \beta_{t+1}(x_{t+1})$$

Stima dello stato ottimo //Algoritmo Forward-Backward

- Usando Forward-Backward:

$$p(x_t \mid y) = \frac{1}{Z_t}\alpha_t(x_t)\beta_t(x_t)$$

- misuro la probabilità a posteriori sul singolo stato nascosto
- Possiamo usare la regola MAP (o moda) per la stima

$$\hat{x}_t = \arg\max_{x_t} \; p(x_t \mid y)$$

- E se volessimo trovare la sequenza di stati con la massima probabilità congiunta? -> Algoritmo di Viterbi

Stima della sequenza di stati ottima //Algoritmo di Viterbi

$$\hat{x} = \arg\max_{x} \; p(x_0, x_1, \ldots, x_T \mid y_1, \ldots, y_T)$$

- E' una forma di programmazione dinamica per trovare (ricorsivamente) la probabilità congiunta della sequenza di stati più probabile che ha generato la sequenza di osservazioni

$$\begin{aligned}\gamma_t(x_t) &\triangleq \max_{x_1,\ldots,x_{t-1}} p(x_1, \ldots, x_{t-1}, x_t \mid y_1, \ldots, y_t) \\ &\propto p(y_t \mid x_t) \cdot \left[\max_{x_{t-1}} \; p(x_t \mid x_{t-1})\gamma_{t-1}(x_{t-1})\right]\end{aligned}$$

Classificazione con HMM

- Training: coppie di (sequenze di stati, sequenze di osservazioni)
- Test: predizione di sequenze di stati da sequenze di osservazioni
- Prima bisogna effettuare il learning dei parametri
  - Esempio: stima di Maximum Likelihood con EM

$$(\hat{Q}, \hat{\theta}) = \arg\max_{Q,\theta} \; p(x \mid Q) \prod_{t=1}^{T} p(y_t \mid x_t, \theta)$$

$$Q = \left[q_{ij}\right] = \left[p(x_{t+1} = i \mid x_t = j)\right]$$

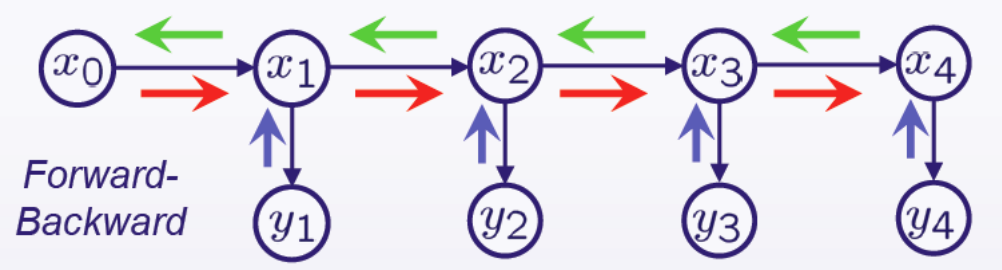
$$\theta = \{\theta_i\}_{i=1}^{N} \qquad \textit{(observation distributions)}$$

- In fase di test, poi si usa Forward-Backward o Viterbi per classificare gli stati

$$C^*(y_{1:T}) = \operatorname*{argmax}_C P(y_{1:T}|C)P(C)$$

Apprendimento dei parametri //Algoritmo di Baum-Welch (EM per HMM)

- Date le sequenze di training
  - Passo E: Fissati i parametri inferisco gli stati nascosti



  - Passo M: Fissati gli stati, aggiorno i parametri di transizione e di osservazione

Apprendimento dei parametri //Algoritmo di Baum-Welch (EM per HMM)

- Date le sequenze di training
  - Passo E: Fissati i parametri inferisco gli stati nascosti

$$\gamma_t(i) \propto \alpha_t(i) .* \beta_t(i) \qquad \gamma_t(i) = P(X_t = i|Y_{1:T})$$

$$\xi_t(i,j) = \frac{\alpha_t(i)a_{ij}b_j(y_{t+1})\beta_{t+1}(j)}{\sum_i \sum_j \alpha_t(i)a_{ij}b_j(y_{t+1})\beta_{t+1}(j)} \qquad \xi_t(i,j) = P(X_t = i, X_{t+1} = j|Y_{1:T})$$

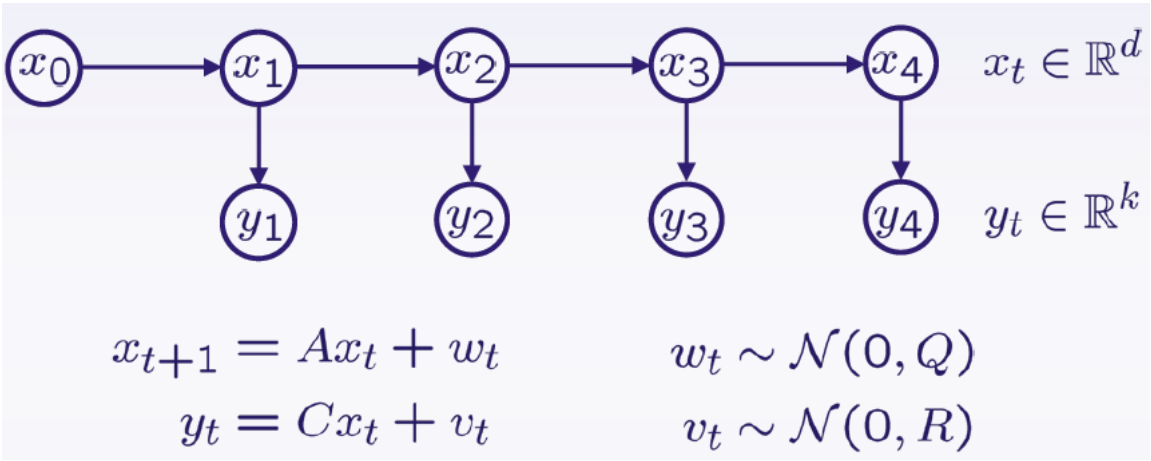
  - Passo M: Fissati gli stati, aggiorno i parametri di transizione e di osservazione

$$\tilde{\pi}(i) = \gamma_1(i)$$ ■ Prior

$$\tilde{a}(i,j) = \frac{\sum_t \xi_t(i,j)}{\sum_t \gamma_t(i)}$$ ■ Matrice di transizione

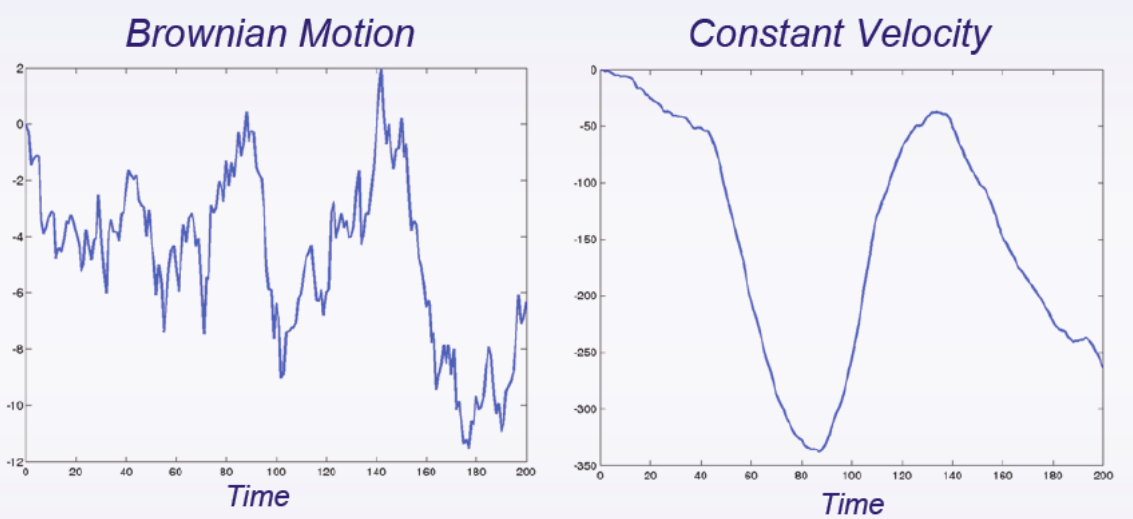
$$\tilde{b}_i(k) = \frac{\sum_{t,y_t=k} \gamma_t(j)}{\sum_t \gamma_t(i)}$$ ■ Matrice di osservazione

HMM continui //Modelli a spazi degli stati lineari



- Il filtro di Kalman è un esempio di HMM continuo

HMM continui
//Modelli a spazi degli stati (lineari)



$$x_{t+1} = x_t + w_t \qquad \begin{bmatrix} x_{t+1} \\ \delta_{t+1} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_t \\ \delta_t \end{bmatrix} + w_t$$

HMM continui
//Filtro di Kalman

$$\begin{aligned} x_{t+1} &= Ax_t + w_t & w_t &\sim \mathcal{N}(0, Q) \\ y_t &= Cx_t + v_t & v_t &\sim \mathcal{N}(0, R) \end{aligned}$$

$$\begin{aligned} p(x_t \mid y_1, \ldots, y_{t-1}) &= \mathcal{N}(x; \tilde{\mu}_t, \tilde{\Lambda}_t) \\ p(x_t \mid y_1, \ldots, y_t) &= \mathcal{N}(x; \mu_t, \Lambda_t) \end{aligned}$$

predizione

osservazione e update

- Utilizzando un modello Gaussiano tutto può essere rappresentato in termini di medie e covarianze

Prediction:
$$\begin{aligned} \tilde{\mu}_t &= A\mu_{t-1} \\ \tilde{\Lambda}_t &= A\Lambda_{t-1}A^T + Q \end{aligned}$$

Kalman Gain:
$$K_t = \tilde{\Lambda}_t C^T (C\tilde{\Lambda}_t C^T + R)^{-1}$$

Update:
$$\begin{aligned} \mu_t &= \tilde{\mu}_t + K_t(y_t - C\tilde{\mu}_t) \\ \Lambda_t &= \tilde{\Lambda}_t - K_t C\tilde{\Lambda}_t \end{aligned}$$

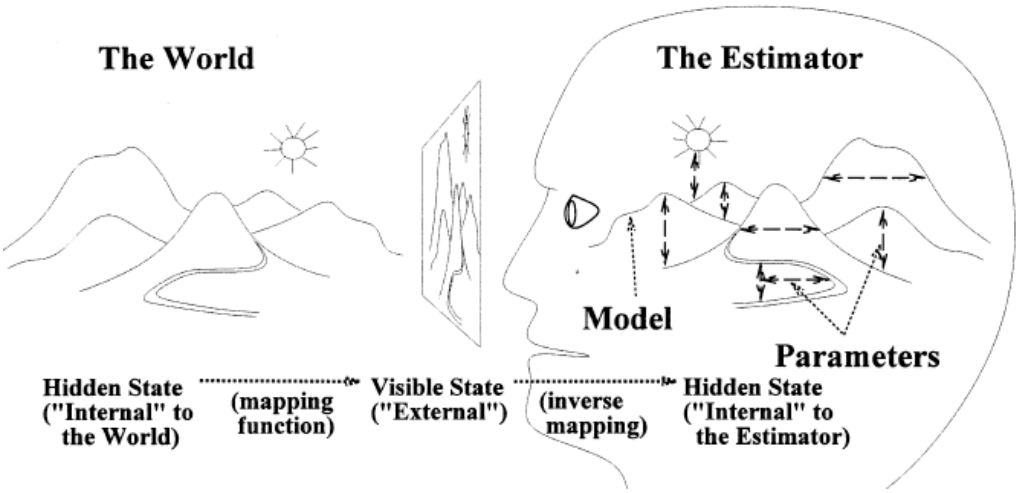
Differenza fra predizione e osservazione

HMM continui
//Filtro di Kalman

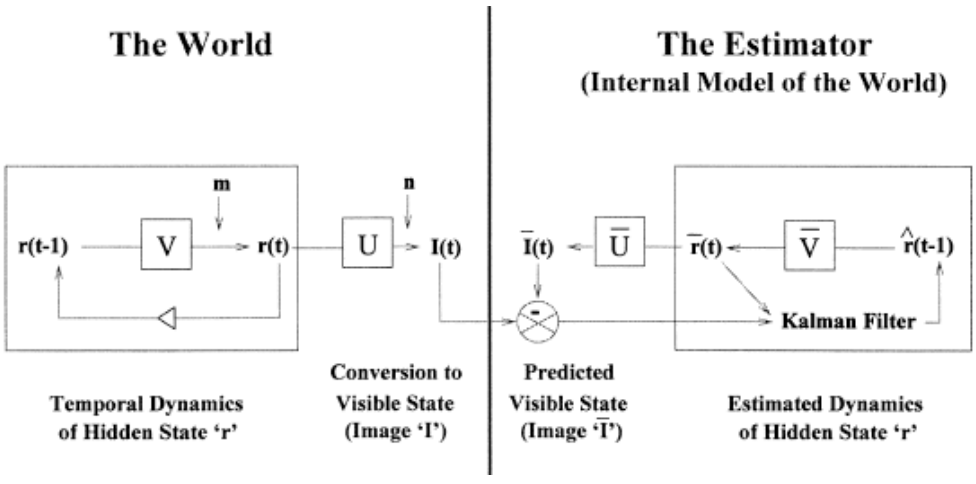
An optimal estimation approach to visual perception and learning☆

Rajesh P.N. Rao *

Vision Research 39 (1999) 1963–1989



HMM continui
//Filtro di Kalman

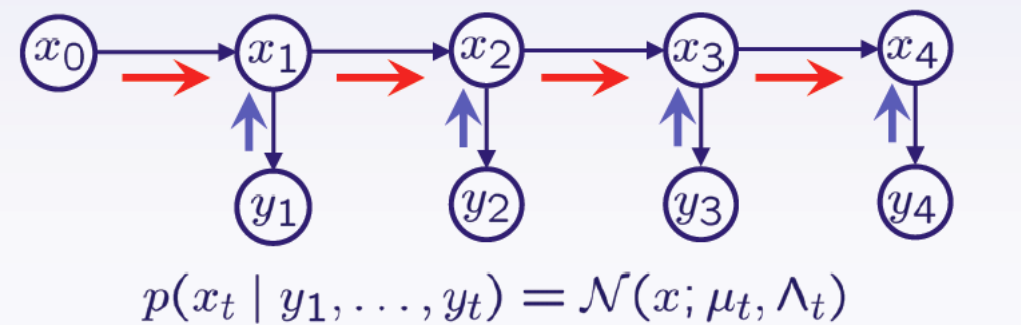


Prediction: $\tilde{\mu}_t = A\mu_{t-1}$, $\tilde{\Lambda}_t = A\Lambda_{t-1}A^T + Q$

Kalman Gain: $K_t = \tilde{\Lambda}_t C^T (C\tilde{\Lambda}_t C^T + R)^{-1}$

Update: $\mu_t = \tilde{\mu}_t + K_t(y_t - C\tilde{\mu}_t)$, $\Lambda_t = \tilde{\Lambda}_t - K_t C\tilde{\Lambda}_t$

Filtro di Kalman come regressore on-line

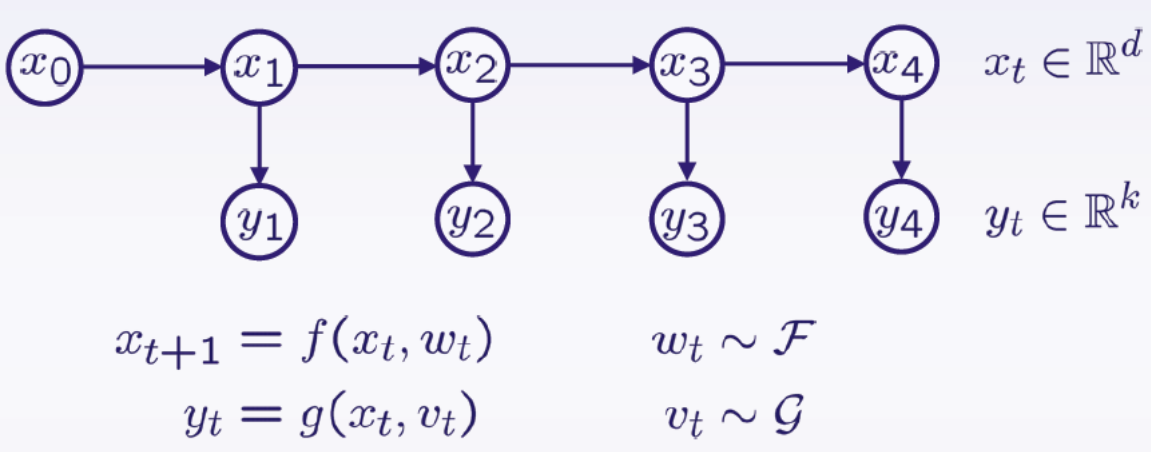


- La media a posteriori minimizza l'errore quadratico medio di predizione

$$\mu_t = \arg\min_{\mu} \; \mathbb{E}\left[\, ||x_t - \mu||^2 \mid y_1, \ldots, y_t \right]$$

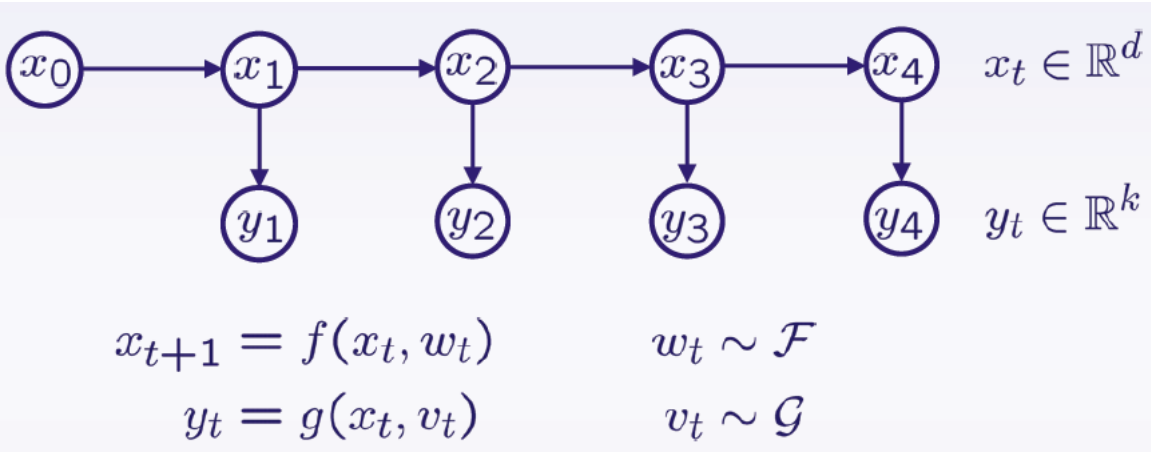
Filtro di Kalman come regressore on-line

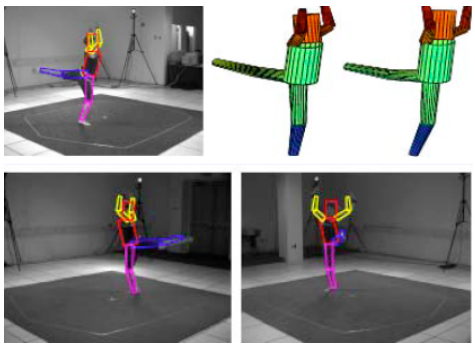
HMM continui
//Modelli a spazi degli stati non lineari



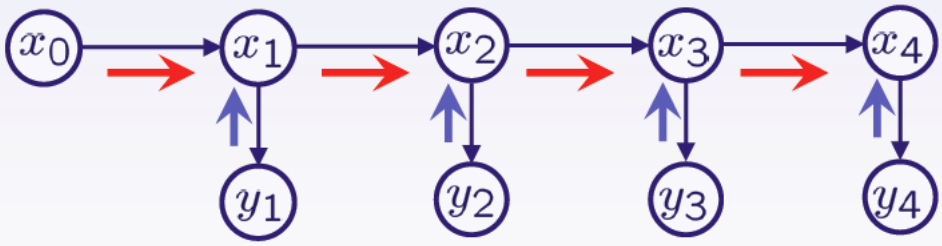
- Dinamica degli stati e osservazioni sono funzioni non lineari (non Gaussiane)

HMM continui
//Modelli a spazi degli stati non lineari





HMM continui
//Filtraggi non lineari



$$p(x_t \mid y_1, \ldots, y_{t-1}) = \tilde{q}_t(x_t)$$

$$p(x_t \mid y_1, \ldots, y_t) = q_t(x_t)$$

Prediction:

$$\tilde{q}_t(x_t) = \int p(x_t \mid x_{t-1}) q_{t-1}(x_{t-1})\, dx_{t-1}$$

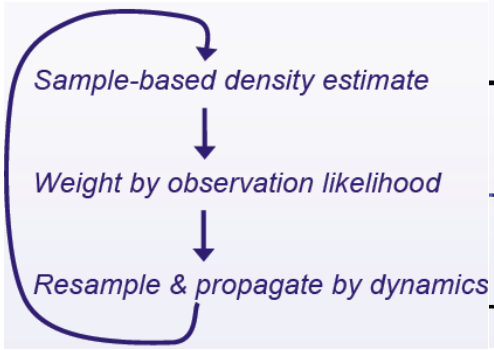
Update:

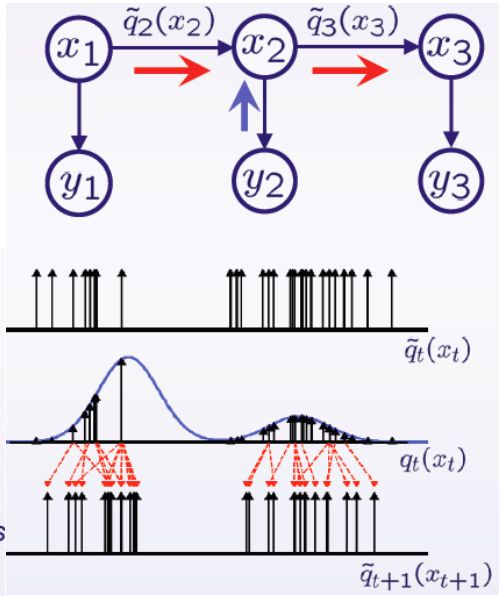
$$q_t(x_t) = \frac{1}{Z_t} \tilde{q}_t(x_t) p(y_t \mid x_t)$$

Filtraggi non lineari approssimati:
//Particle filtering (Condensation, etc...)

$$q_t(x_t) \propto p(y_t \mid x_t) \cdot \int p(x_t \mid x_{t-1}) q_{t-1}(x_{t-1})\, dx_{t-1}$$

Le distribuzioni di probabilità vengono rappresentati con dei campioni





Filtraggi non lineari approssimati: //Particle filtering (Condensation, etc...)

Sequential importance sampling step

– For $i = 1, ..., N$, sample from the transition priors

$$\widetilde{x}_t^{(i)} \sim q_t\left(\widetilde{x}_t | x_{0:t-1}^{(i)}, y_{1:t}\right)$$

and set

$$\widetilde{x}_{0:t}^{(i)} \triangleq \left(\widetilde{x}_t^{(i)}, x_{0:t-1}^{(i)}\right)$$

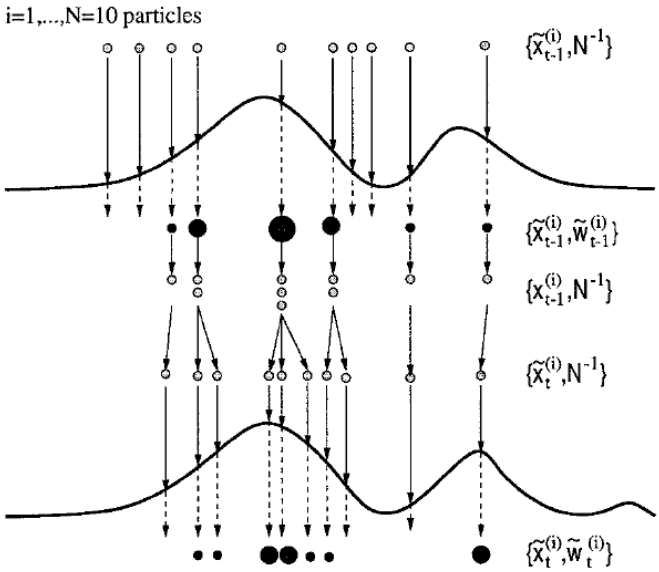
– For $i = 1, ..., N$, evaluate and normalize the importance weights

$$w_t^{(i)} \propto \frac{p\left(y_t | \widetilde{x}_t^{(i)}\right) p\left(\widetilde{x}_t^{(i)} | x_{0:t-1}^{(i)}, y_{1:t-1}\right)}{q_t\left(\widetilde{x}_t^{(i)} | x_{0:t-1}^{(i)}, y_{1:t}\right)}.$$

Selection step

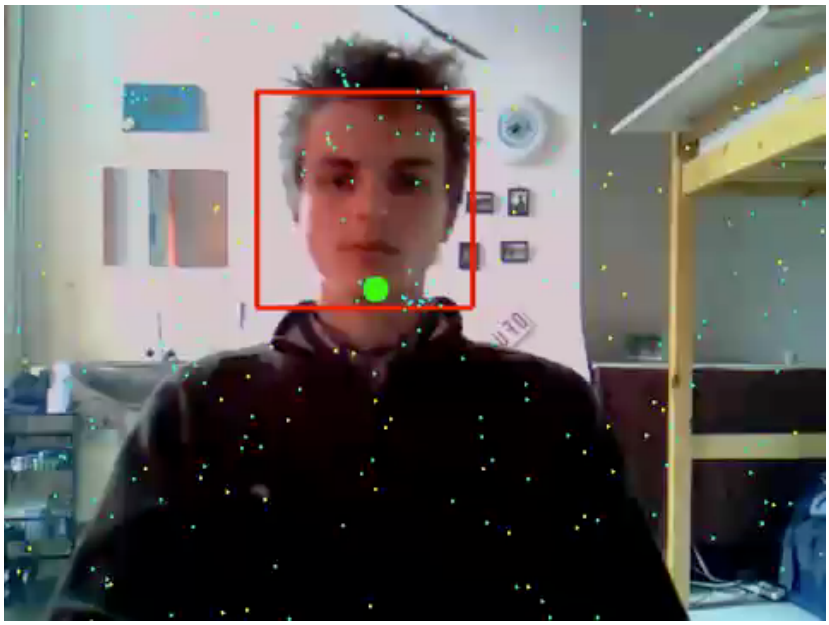
– Multiply/Discard particles $\left\{\widetilde{x}_{0:t}^{(i)}\right\}_{i=1}^{N}$ with high/low importance weights $w_t^{(i)}$ to obtain N particles $\left\{x_{0:t}^{(i)}\right\}_{i=1}^{N}$.

Filtraggi non lineari approssimati: //Particle filtering (Condensation, etc...)



filter starts at time $t-1$ with an unweighted measure $\{\tilde{\mathbf{x}}_{t-1}^{(i)}, N^{-1}\}$, which provides an approximation of $p(\mathbf{x}_{t-1} \mid y_{1:t-2})$. For each particle we compute the importance weights using the information at time $t-1$. This results in the weighted measure $\{\tilde{\mathbf{x}}_{t-1}^{(i)}, \tilde{w}_{t-1}^{(i)}\}$, which yields an approximation $p(\mathbf{x}_{t-1} \mid y_{1:t-1})$. Subsequently, the resampling step selects only the "fittest" particles to obtain the unweighted measure $\{\mathbf{x}_{t-1}^{(i)}, N^{-1}\}$, which is still an approximation of $p(\mathbf{x}_{t-1} \mid y_{1:t-1})$. Finally, the sampling (prediction) step introduces variety, resulting in the measure $\{\tilde{\mathbf{x}}_t^{(i)}, N^{-1}\}$, which is an approximation of $p(\mathbf{x}_t \mid y_{1:t-1})$.

Filtraggi non lineari approssimati: //Particle filtering (Condensation, etc...)



Generalizzazioni degli HMM //Dynamic Bayesian Network