Machine Learning for Data Science (CS4786)
Lecture 24

HMM Particle Filter

Course Webpage :
http://www.cs.cornell.edu/Courses/cs4786/2017fa/

Rejection Sampling

Rejection Sampling

- Sample variables from joint distribution (both latent and observed variables)

Rejection Sampling

- Sample variables from joint distribution (both latent and observed variables)
- Throw away all ones that don’t match observation

Rejection Sampling

- Sample variables from joint distribution (both latent and observed variables)
- Throw away all ones that don’t match observation
- Now compute empirical frequencies

Rejection Sampling

- Sample variables from joint distribution (both latent and observed variables)
- Throw away all ones that don't match observation
- Now compute empirical frequencies

Problem: too wasteful (too many Rejections)

Rejection Sampling

Algorithm:

Topologically sort variables (parents first children later)

For $t = 1$ to n (number of samples)

For $i = 1$ to N (number of variables in model)

Sample $x_i^t \sim P(X_i|\text{Parents}(X_i)\text{ already sampled})$

End For

End For

Discard samples that do not match observations

Compute empirical frequencies

Importance Sampling

- We really want to draw from distribution P.
- But we can only draw from distribution Q easily
- Trick:
  - Draw $x_1, \ldots, x_n \sim Q$
  - Re-weight each sample x_t by $P(X = x_t)/Q(X = x_t)$

Importance Sampling

Importance Sampling

- For bayesian networks:

Importance Sampling

- For bayesian networks:
  - What we want:

$$P(X_{\text{Latent}_1}, \ldots, X_{\text{Latent}_m} | X_{\text{Observed}_1}, \ldots X_{\text{Observed}_n})$$

Importance Sampling

- For bayesian networks:
  - What we want:

$$P(X_{\text{Latent}_1}, \ldots, X_{\text{Latent}_m} | X_{\text{Observed}_1}, \ldots X_{\text{Observed}_n})$$

  - What we sample from:

$$\prod_{j=1}^{m} P(X_{\text{Latent}_j} | \text{Parent}(X_{\text{Latent}_j}))$$

Importance Sampling

- For bayesian networks:
  - What we want:

$$P(X_{\text{Latent}_1}, \ldots, X_{\text{Latent}_m} | X_{\text{Observed}_1}, \ldots X_{\text{Observed}_n})$$

  - What we sample from:

$$\prod_{j=1}^{m} P(X_{\text{Latent}_j} | \text{Parent}(X_{\text{Latent}_j}))$$

  - Weight: $\prod_{i=1}^{n} P(X_{\text{Observed}_i} | \text{Parent}(X_{\text{Observed}_i}))$

Importance Sampling

Likelihood weighting:

Topologically sort variables (parents first children later)

For $t = 1$ to n (number of samples)

Set $w_t = 1$

For $i = 1$ to N (number of variables)

If X_i is observed,

Set $w_t \leftarrow w_t \cdot P(X_i = x_i | \text{Parents}(X_i) = \text{ already sampled})$

Set $x_i^t = x_i$ (the observed value)

Else, sample $x_i^t \sim P(X_i | \text{Parents}(X_i) = \text{ already sampled})$

End For

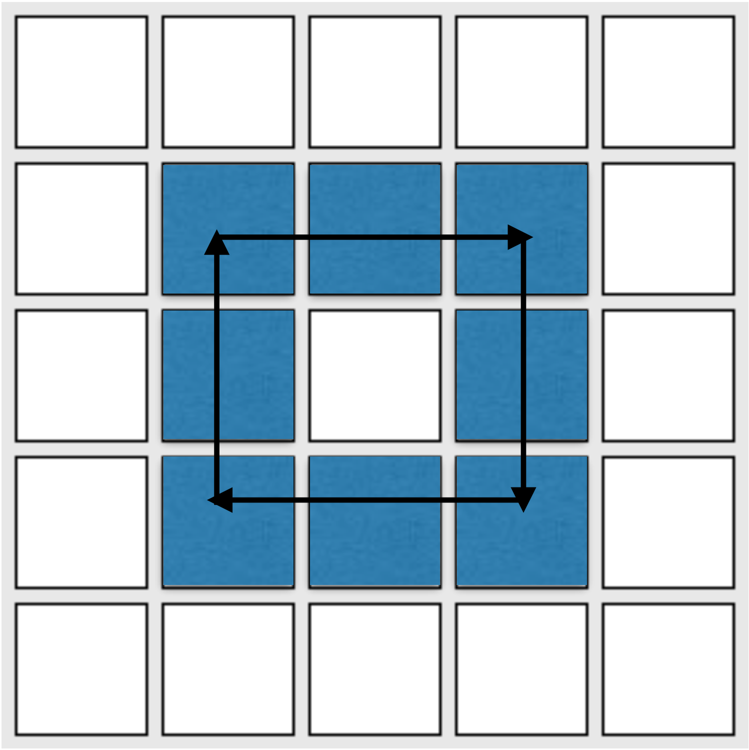
End For

Output,

$$P(\text{Variable} = \text{value} | \text{Observation}) = \frac{\sum_{t=1}^{n} w_t \mathbf{1}\{\text{Variable} = \text{value}\}}{\sum_{t=1}^{n} w_t}$$

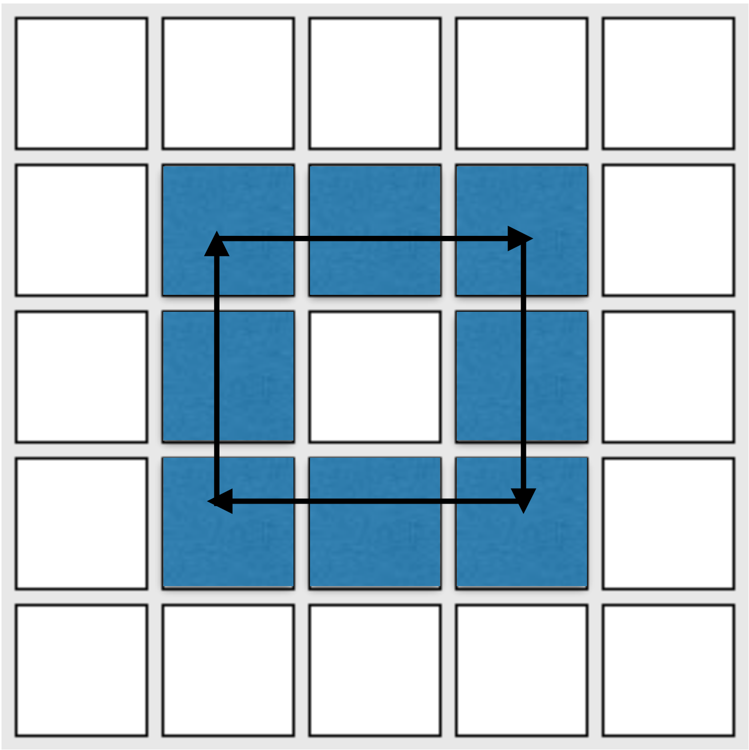
Hidden Markov Model (HMM)

Hidden Markov Model (HMM)



Hidden Markov Model (HMM)

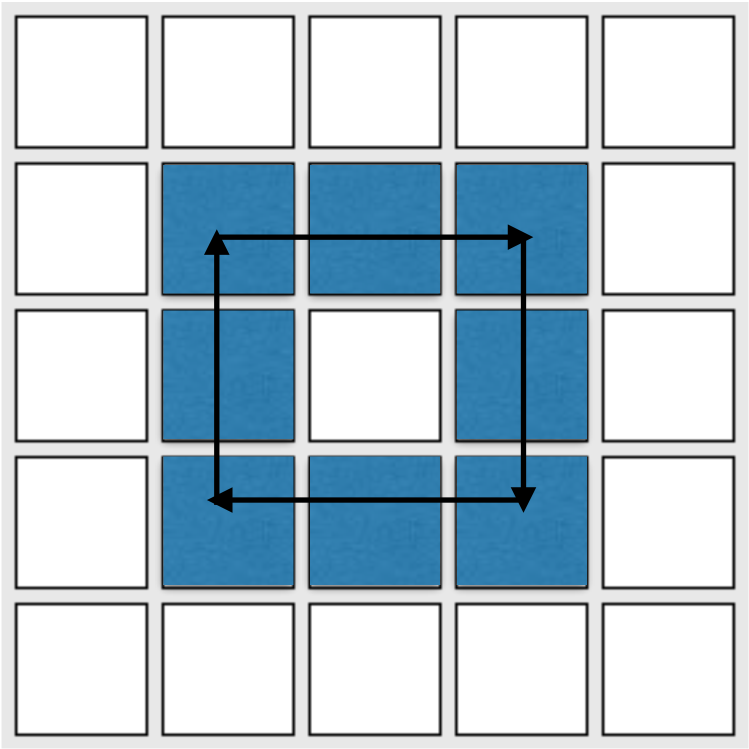
Example:



HIDDEN MARKOV MODEL (HMM)

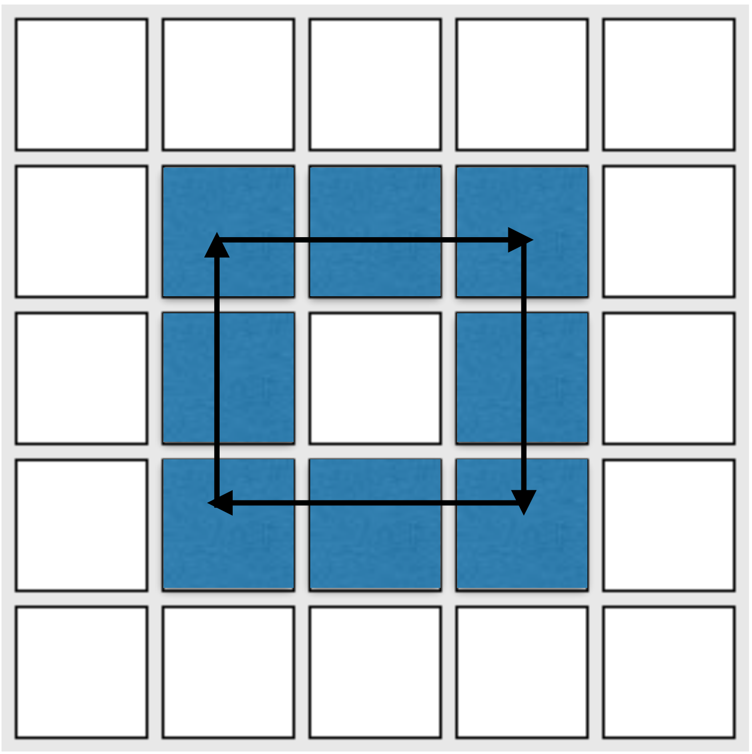
Example:

But you don't observe location (dark room)



Hidden Markov Model (HMM)

Example:

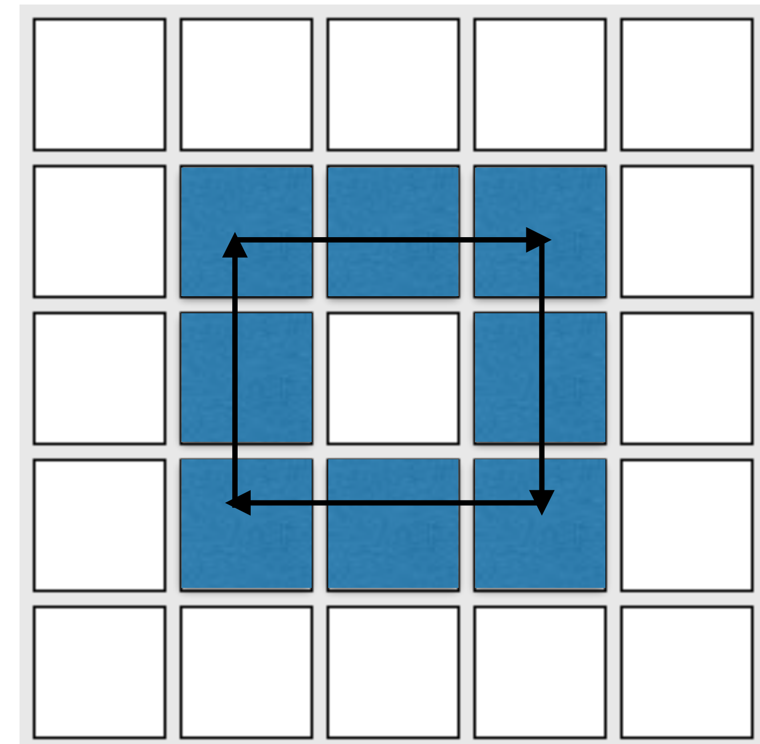


But you don't observe location (dark room)

You hear how close the bot is!

Hidden Markov Model (HMM)

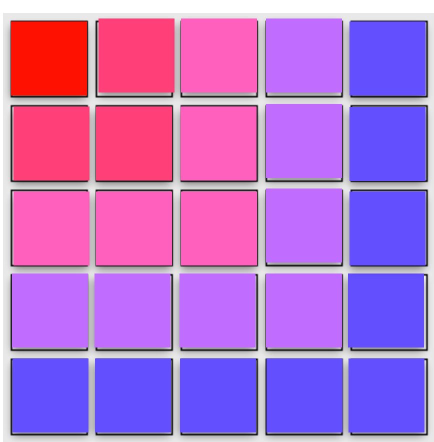
Example:



But you don't observe location (dark room)

You hear how close the bot is!

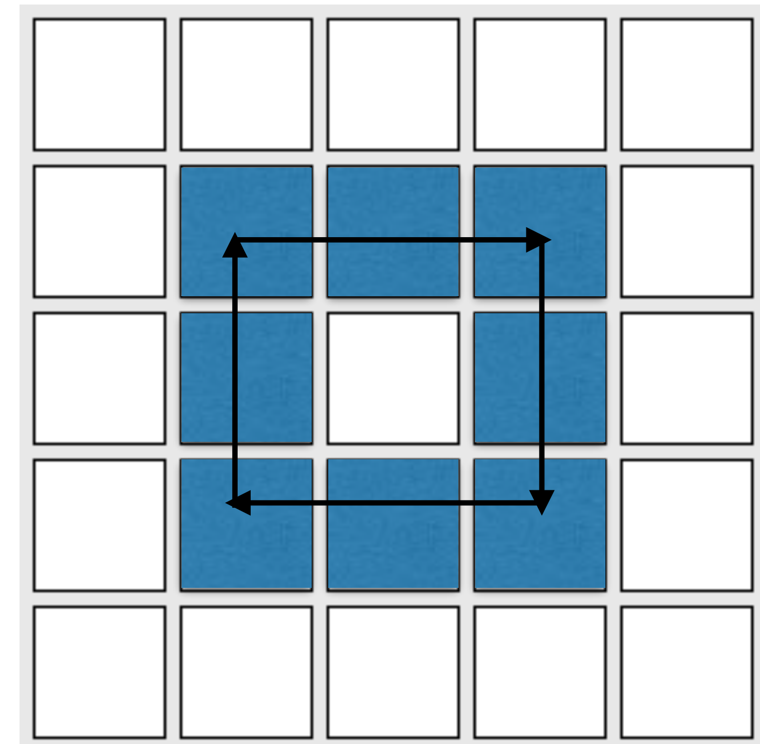
What you hear:



+ noise

Hidden Markov Model (HMM)

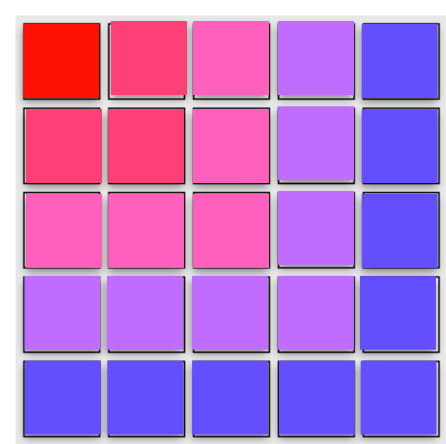
Example:



But you don't observe location (dark room)

You hear how close the bot is!

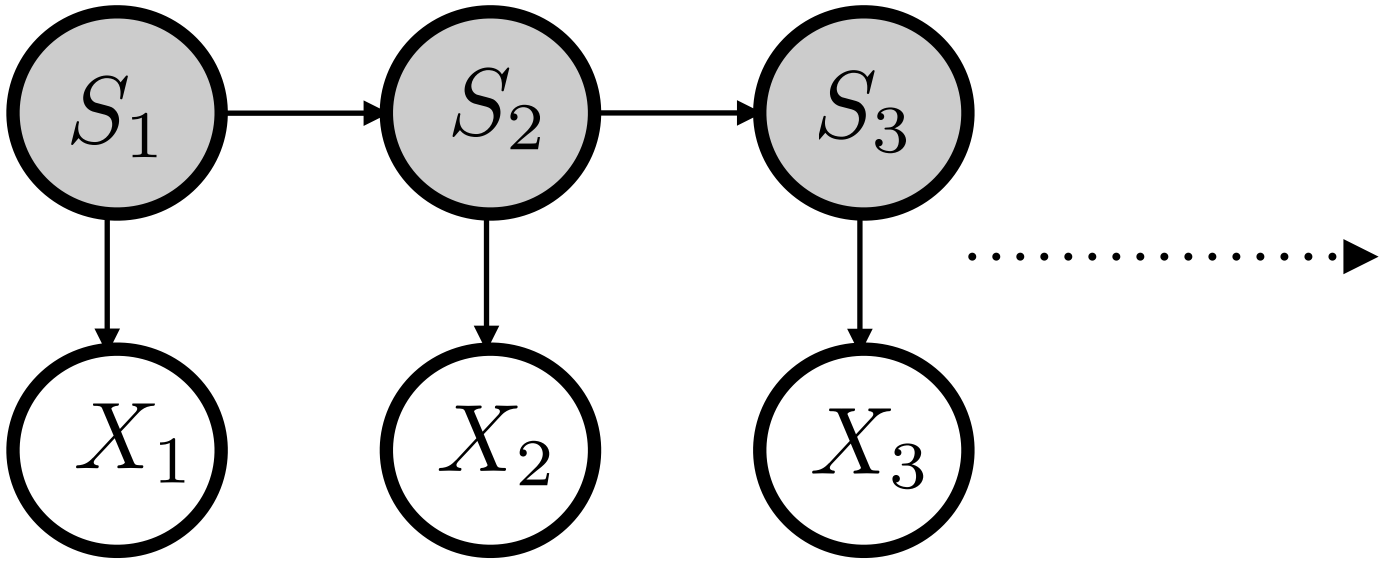
What you hear:



+ noise

Can you catch the Bot? In time?

HIDDEN MARKOV MODEL (HMM)



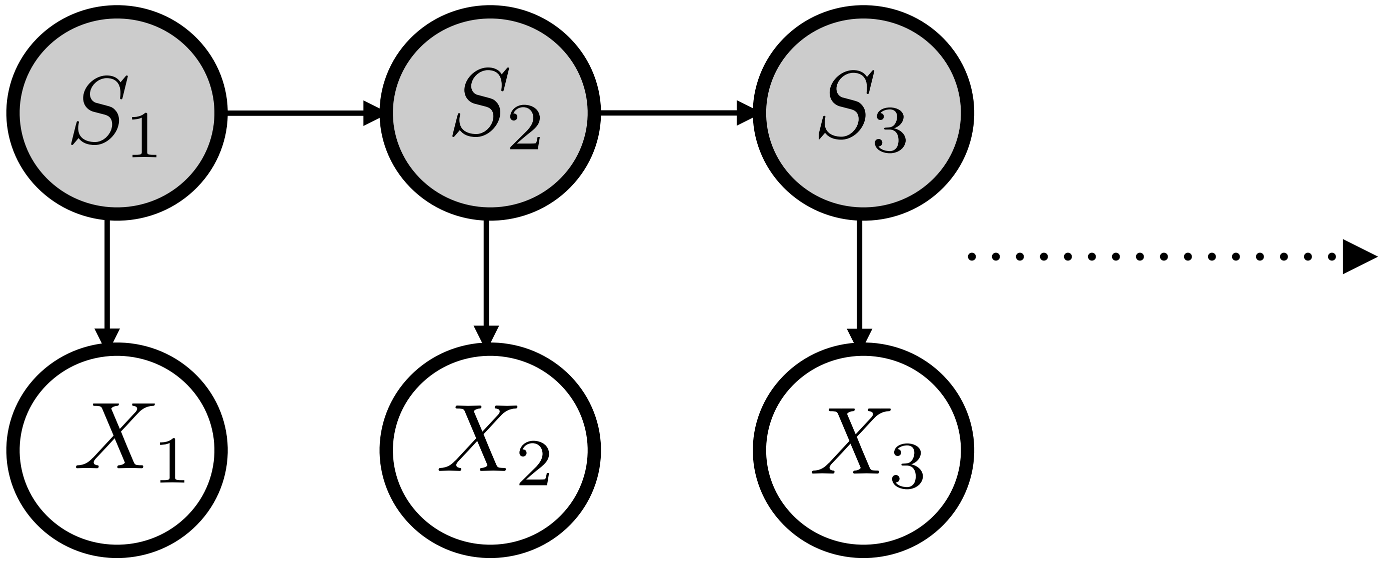
X_t's are what you hear (observation)

S_t's are the unseen locations (states)

Eg: for m x m grid we have, $K = m^2$ states

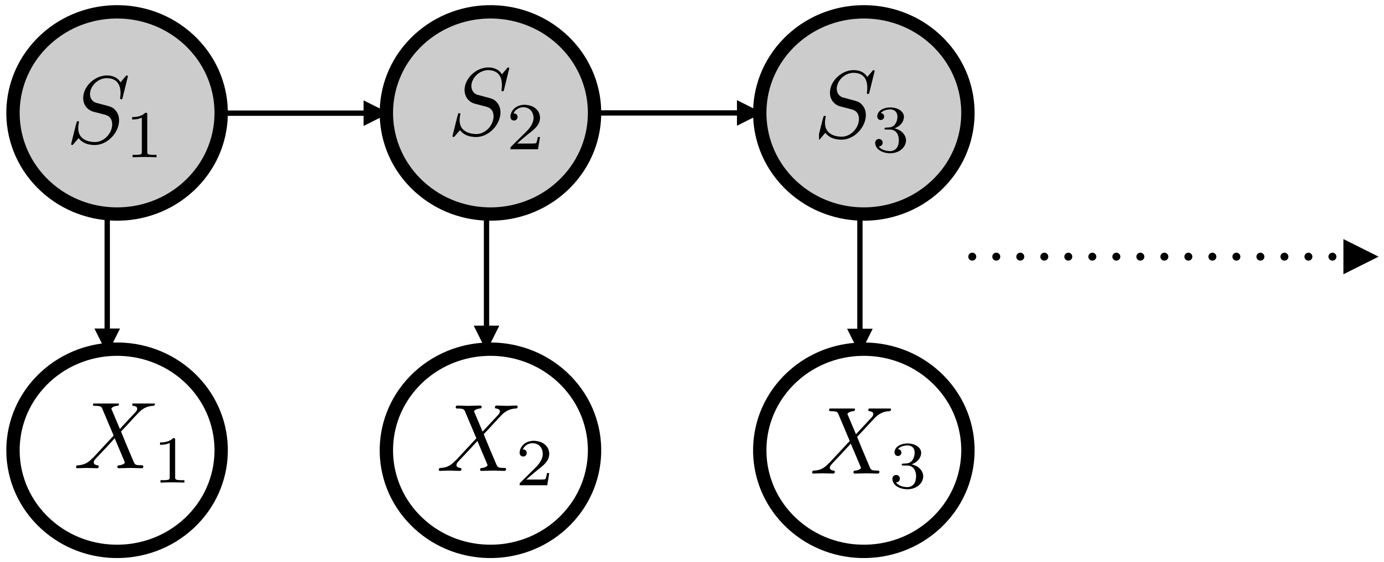
Number of alphabets = # colors you can observe

HIDDEN MARKOV MODEL (HMM)



Eg: for m x m grid we have, K = m^2 states

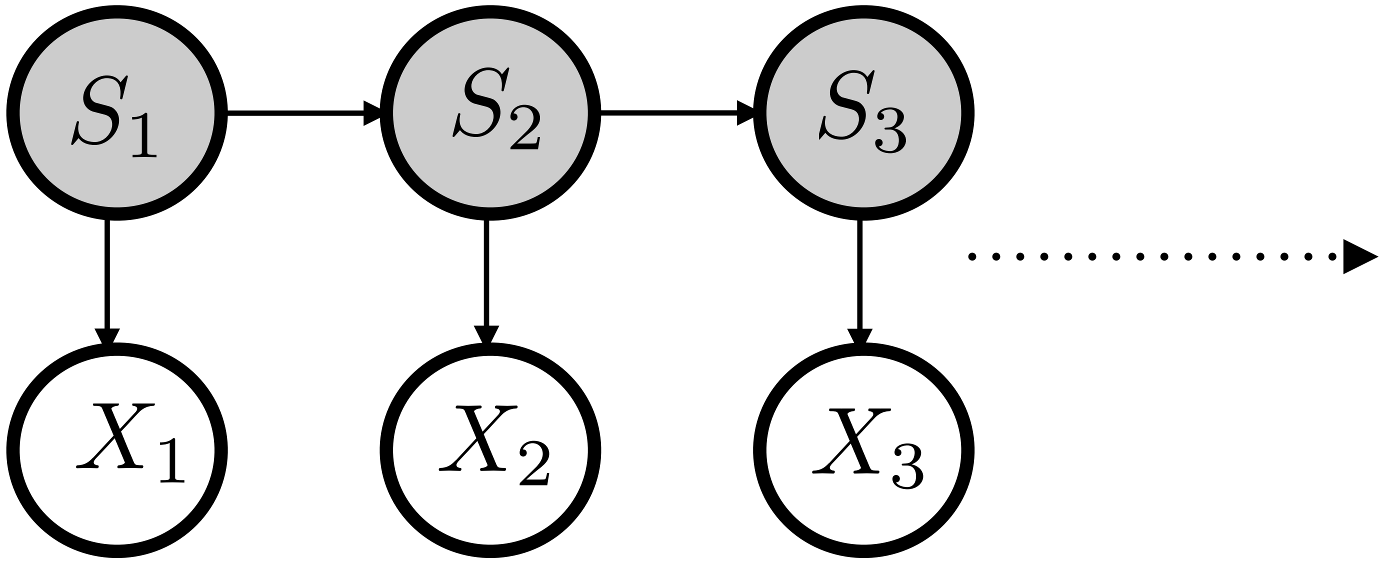
HIDDEN MARKOV MODEL (HMM)



Eg: for m x m grid we have, $K = m^2$ states

Transition matrix is K x K (too large)

Hidden Markov Model (HMM)



Eg: for m x m grid we have, $K = m^2$ states

Transition matrix is K x K (too large)

Use sampling to do approximate inference

Number of samples $n << m^4$

Inference Question

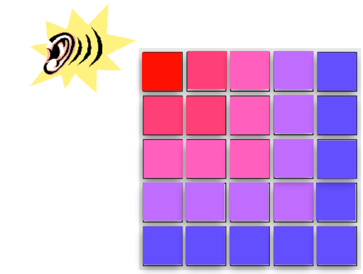
- Can we compute (efficiently and approximately)

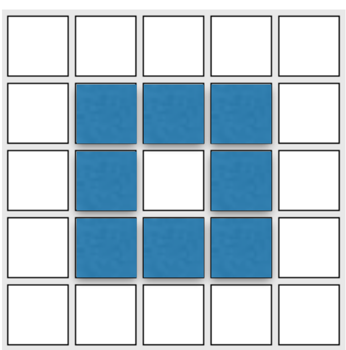
$$P(S_t|x_1, \ldots, x_{t-1})$$

- We cant afford too much time to compute since we need to move the bot in time

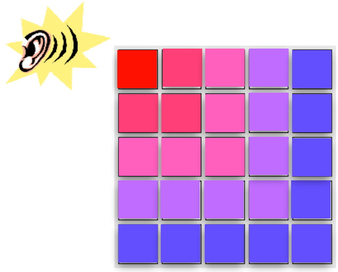
Hidden Markov Model (HMM)

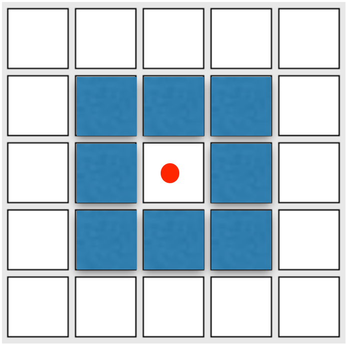
Hidden Markov Model (HMM)



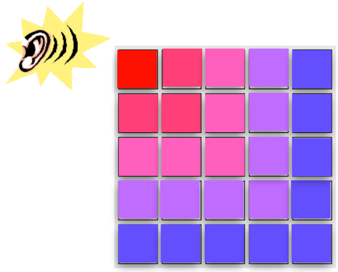


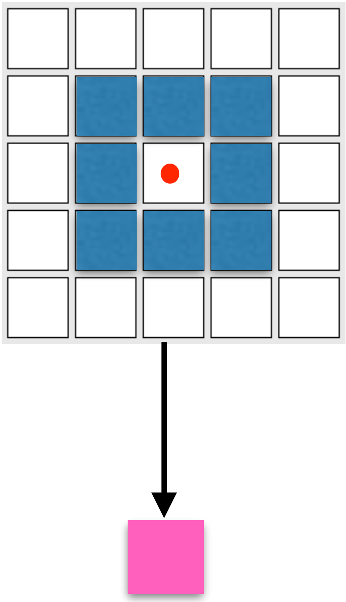
Hidden Markov Model (HMM)



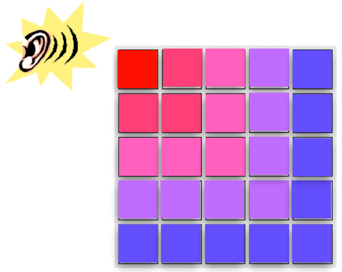


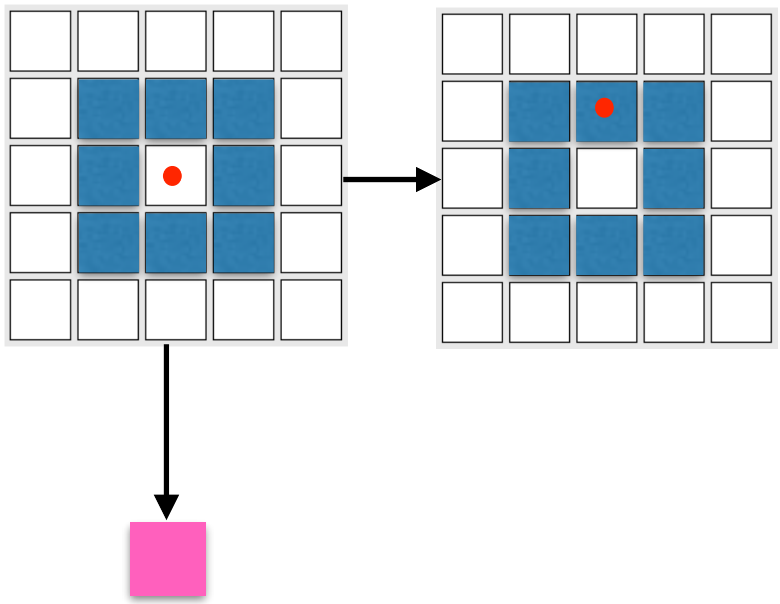
Hidden Markov Model (HMM)



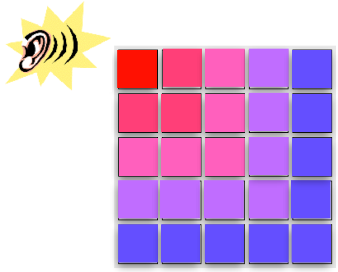


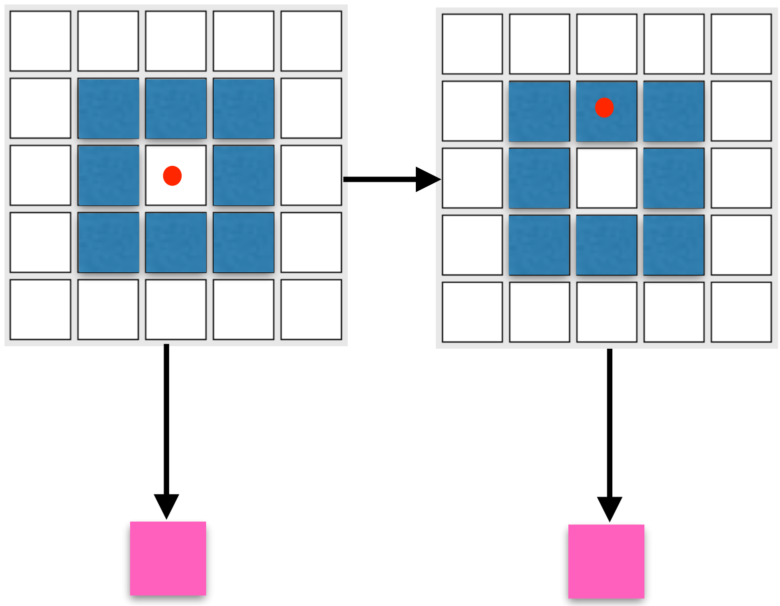
Hidden Markov Model (HMM)



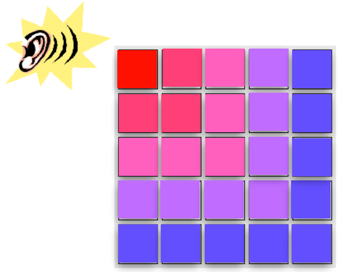


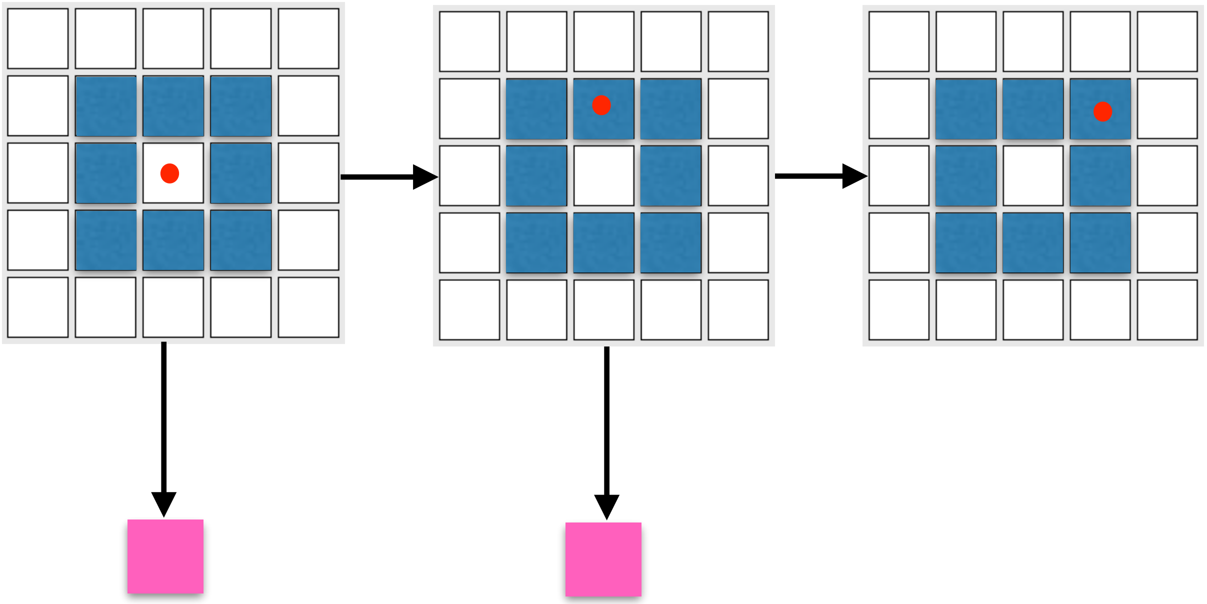
HIDDEN MARKOV MODEL (HMM)



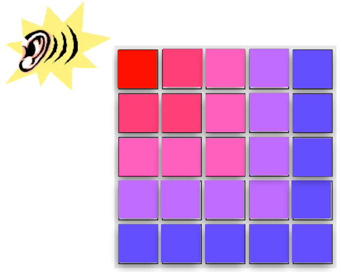


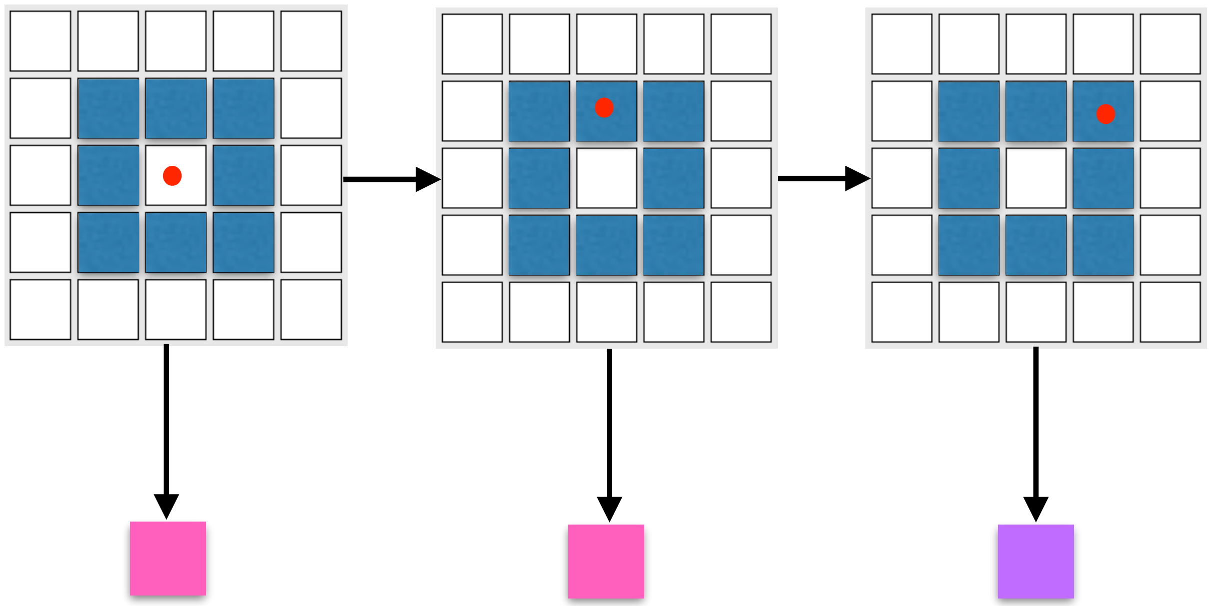
Hidden Markov Model (HMM)



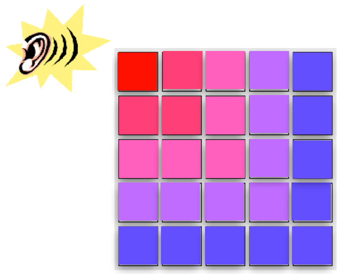


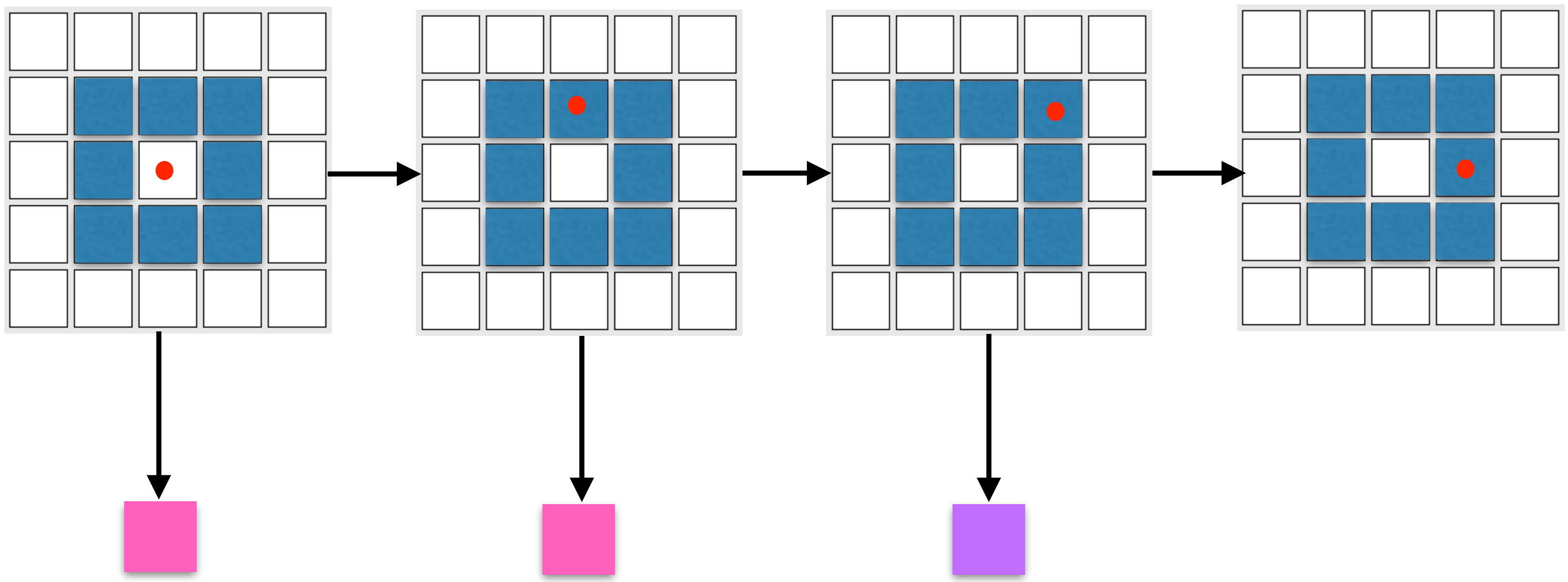
HIDDEN MARKOV MODEL (HMM)



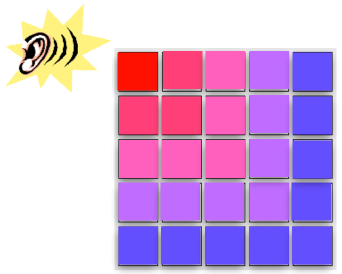


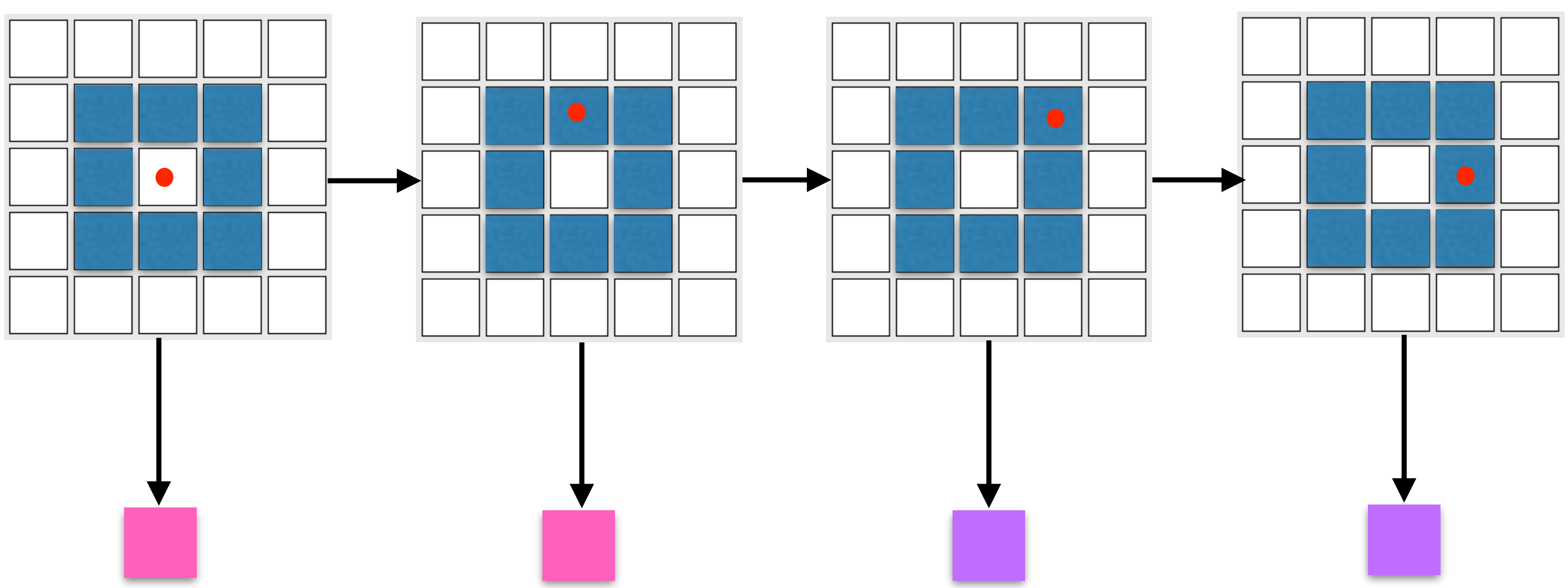
Hidden Markov Model (HMM)



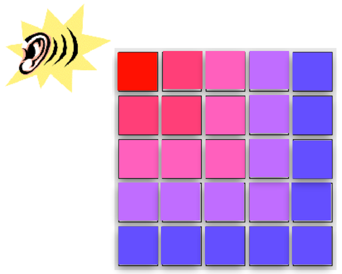


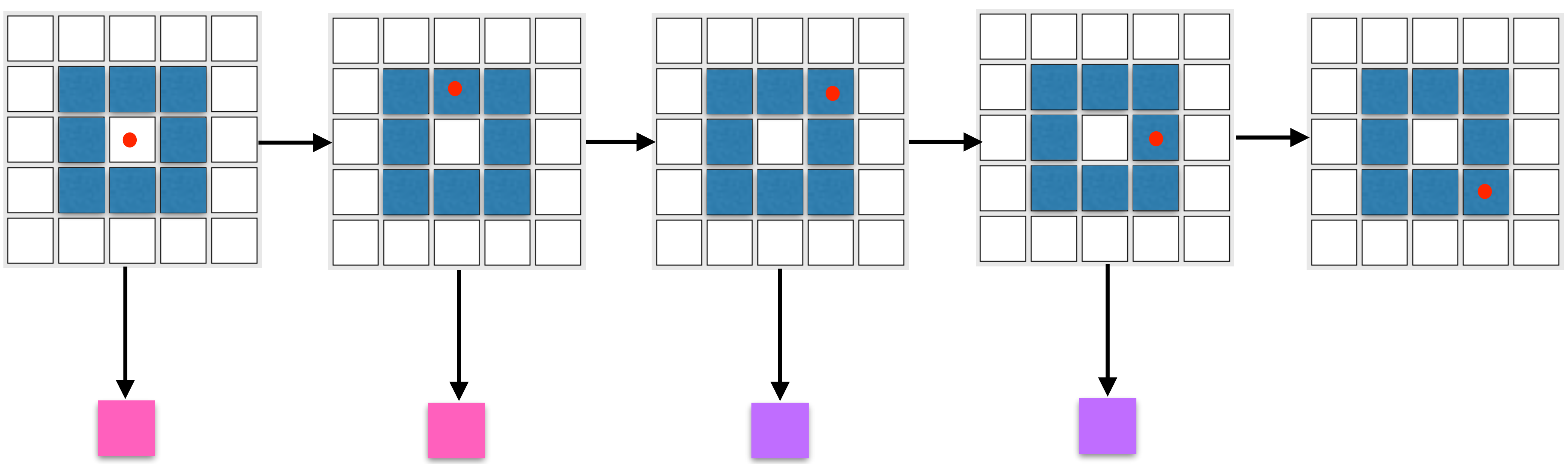
Hidden Markov Model (HMM)



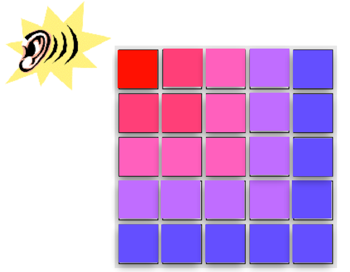


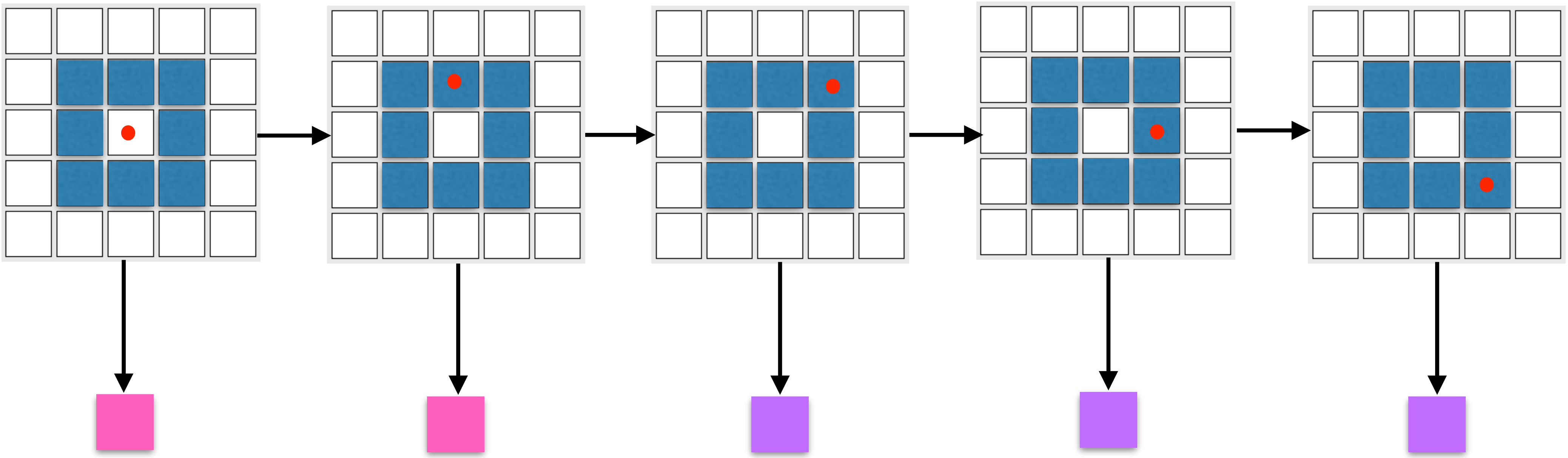
Hidden Markov Model (HMM)



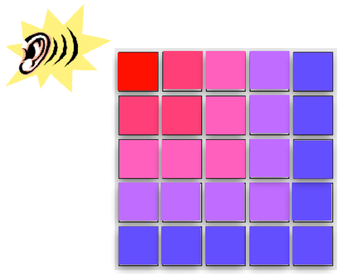


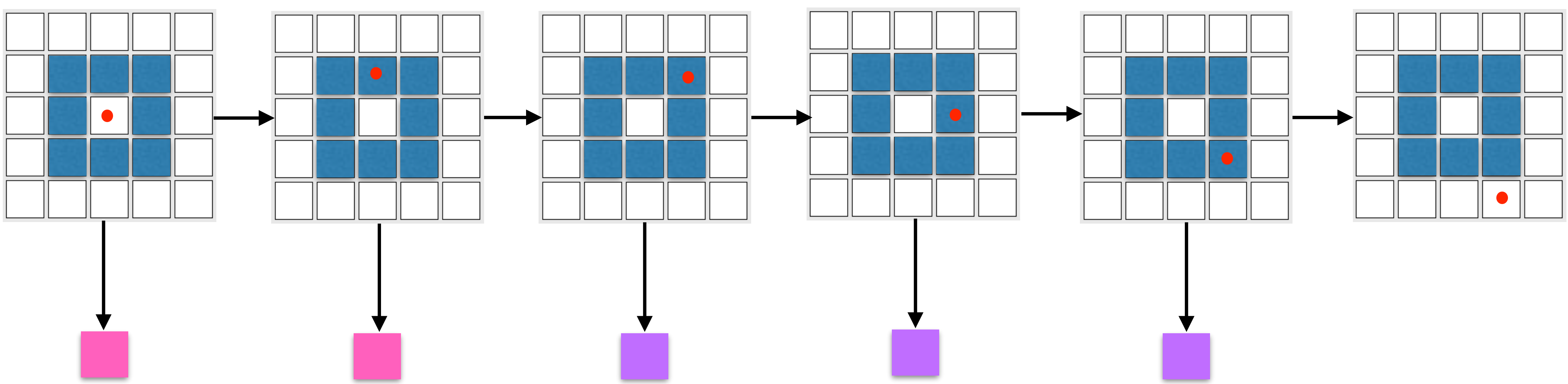
Hidden Markov Model (HMM)



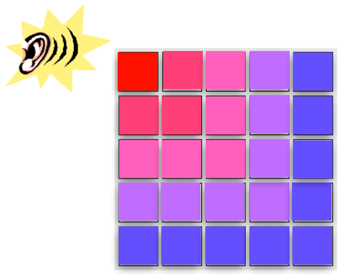


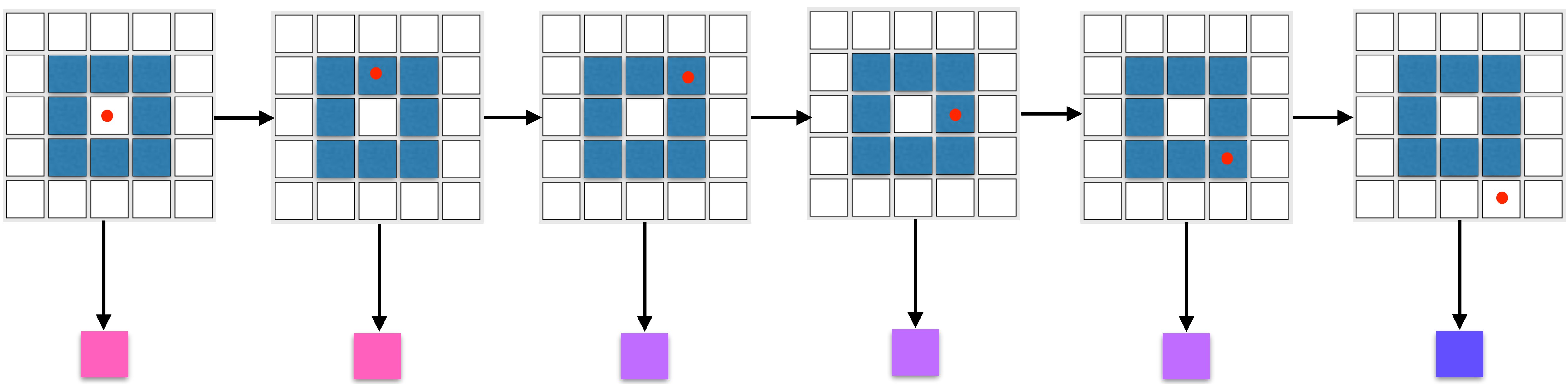
Hidden Markov Model (HMM)



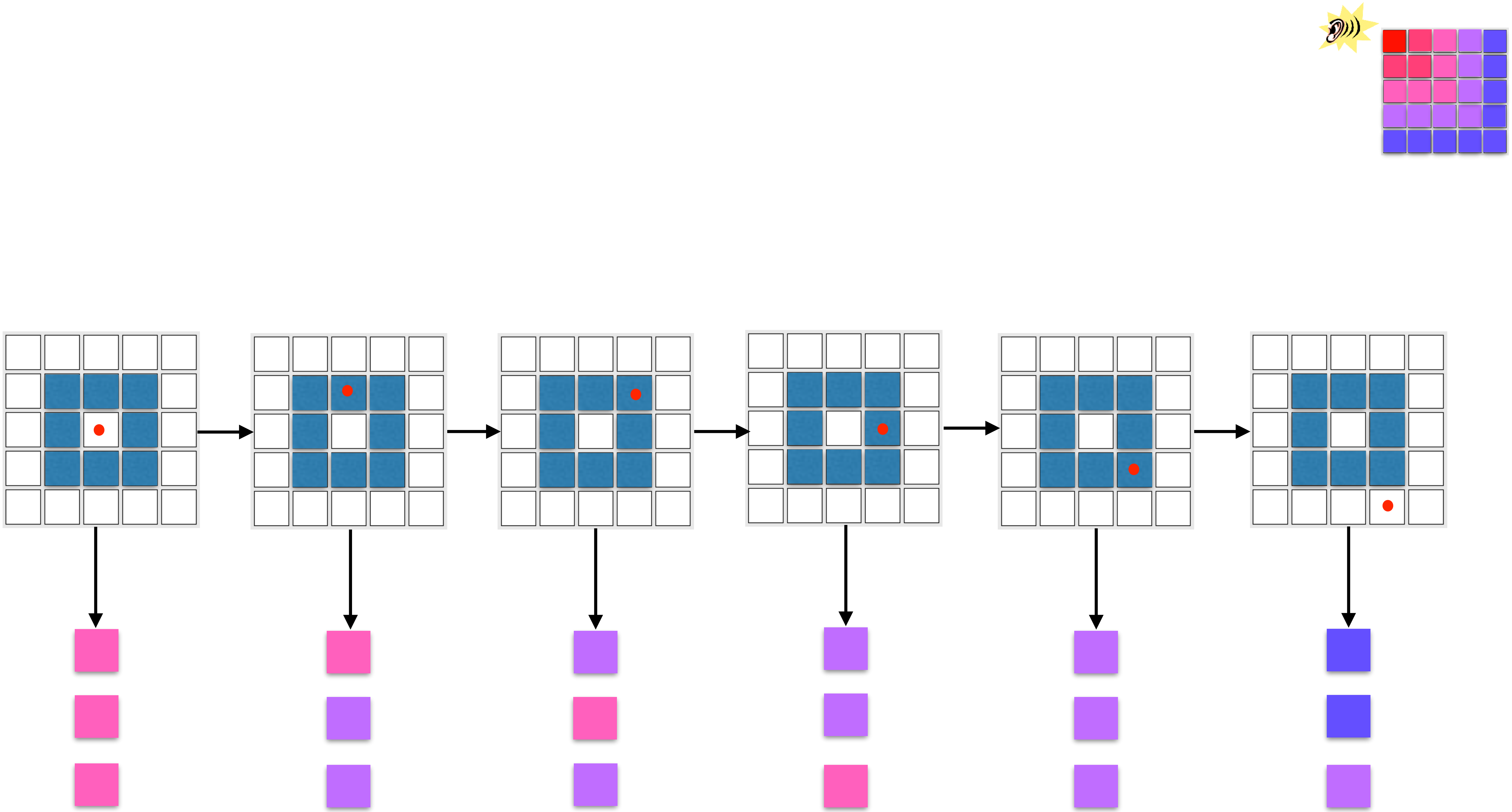


Hidden Markov Model (HMM)

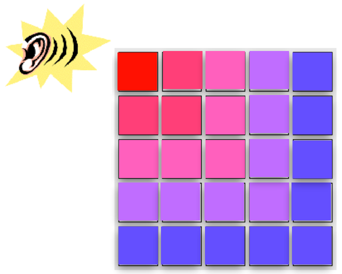




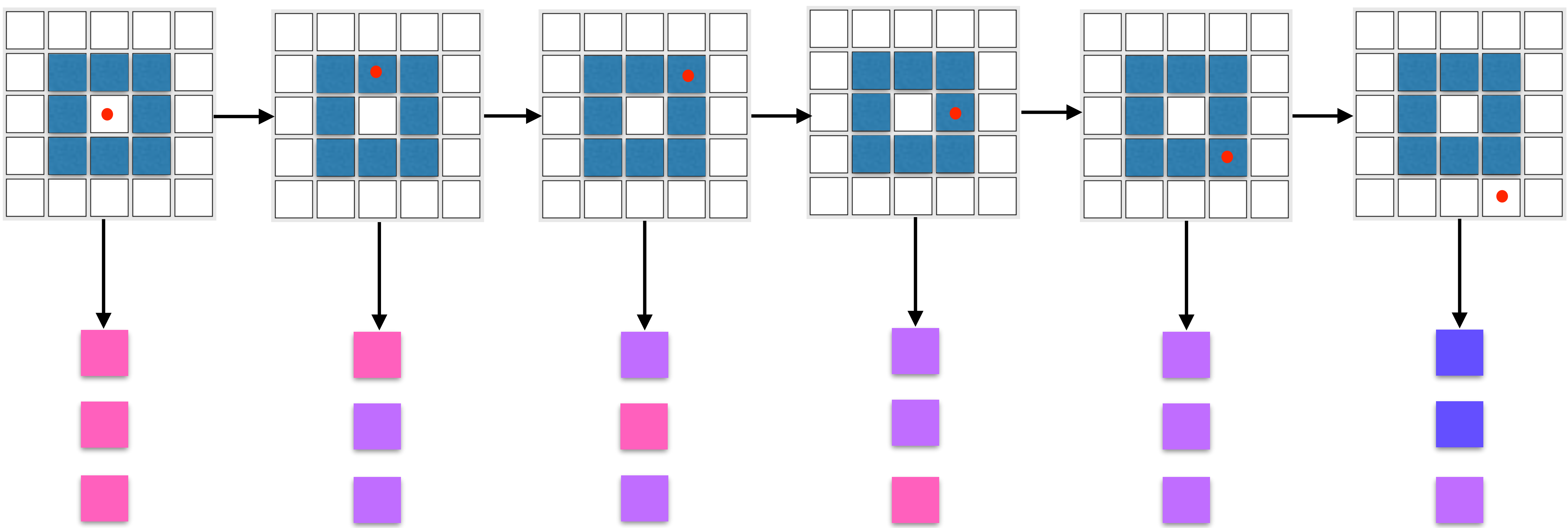
Hidden Markov Model (HMM)



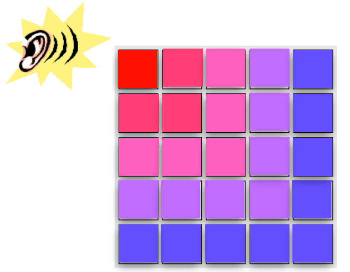
HIDDEN MARKOV MODEL (HMM)



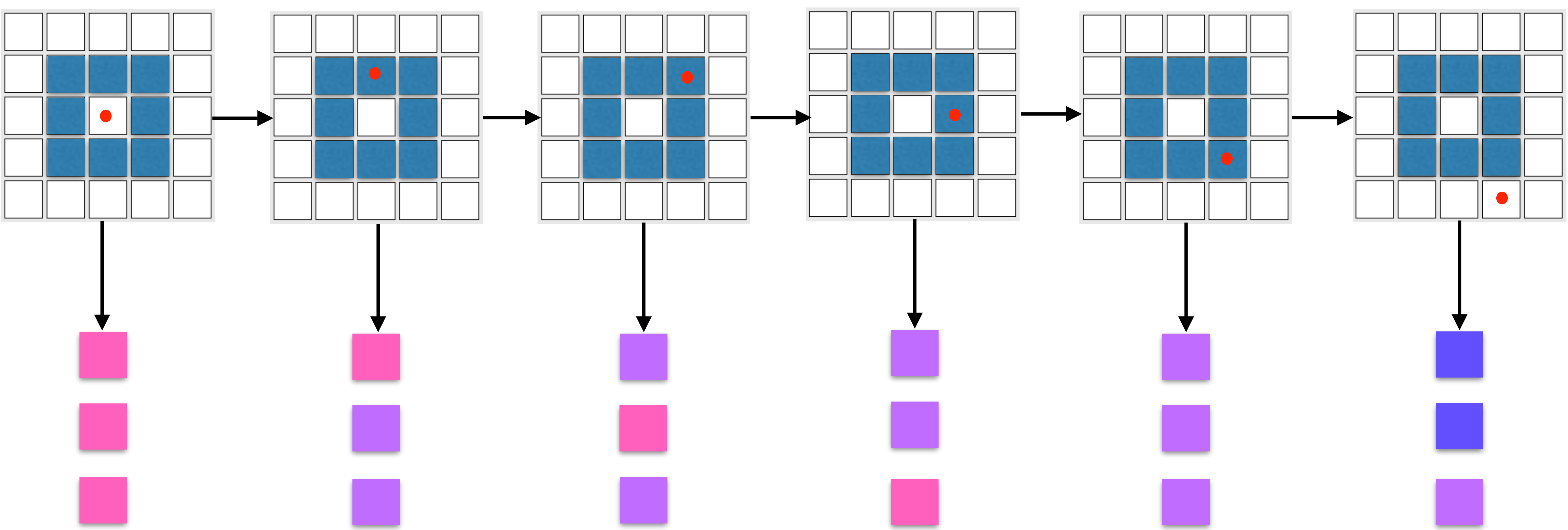
Eg: say observations were



Hidden Markov Model (HMM)

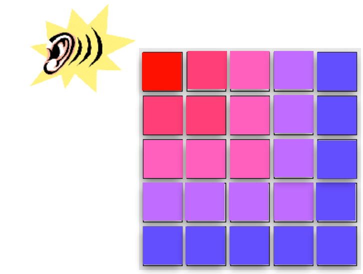


Eg: say observations were

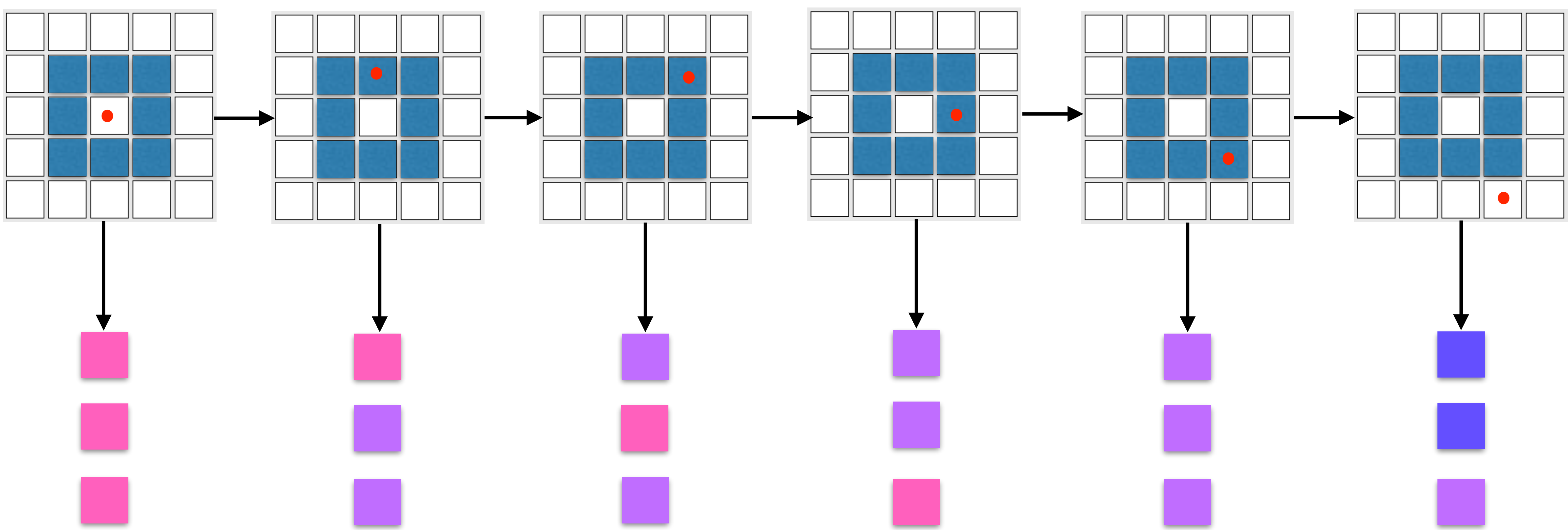


Rejection sampling: Reject samples that don't match observations

Hidden Markov Model (HMM)



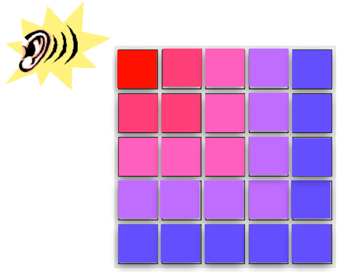
Eg: say observations were



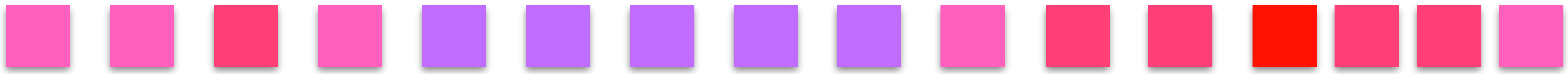
Rejection sampling: Reject samples that don't match observations

We can do this sequentially!

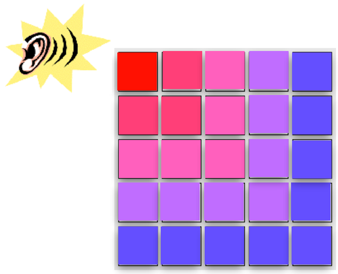
Hidden Markov Model (HMM)



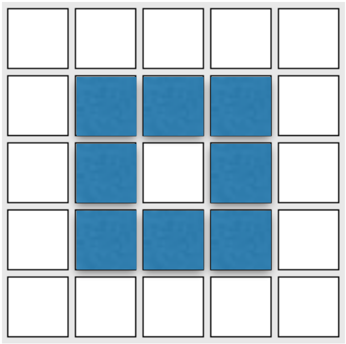
Eg: say observations were



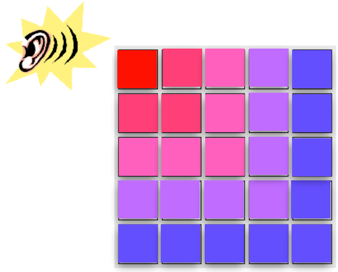
Hidden Markov Model (HMM)



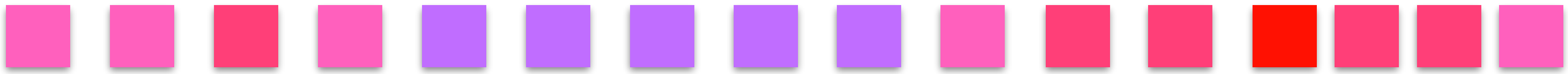
Eg: say observations were

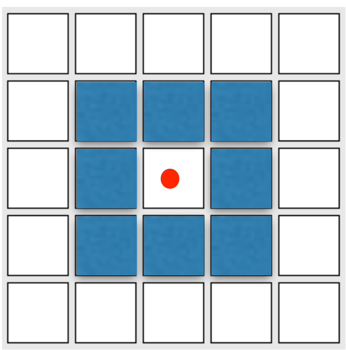


Hidden Markov Model (HMM)

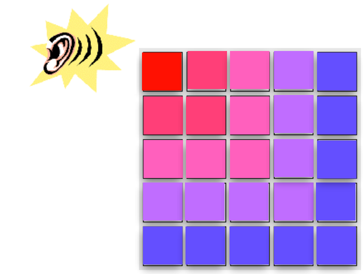


Eg: say observations were

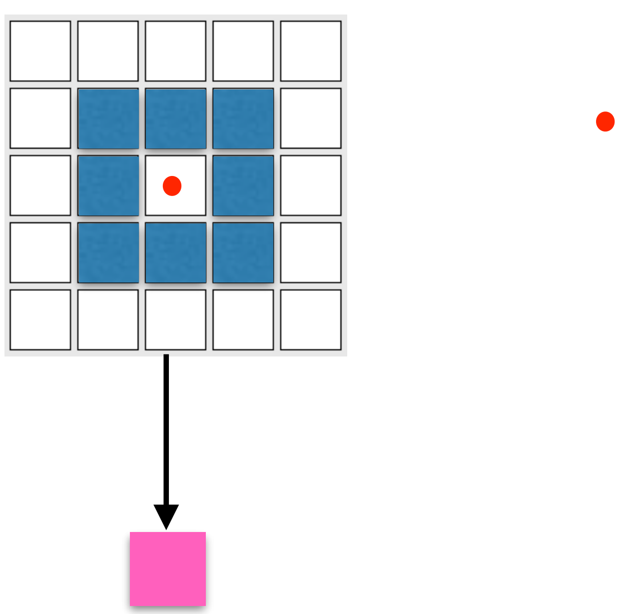




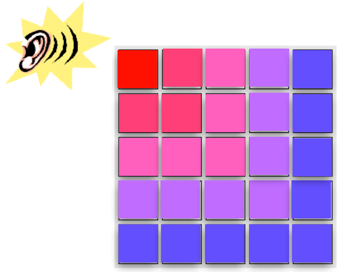
Hidden Markov Model (HMM)



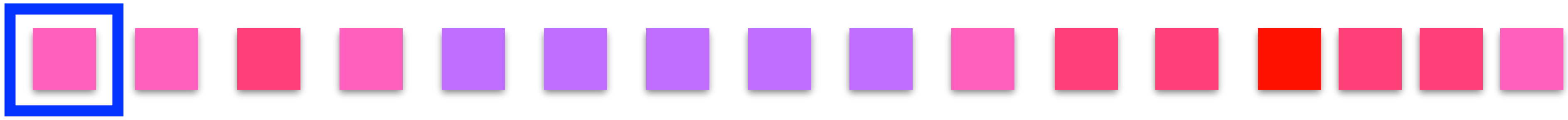
Eg: say observations were

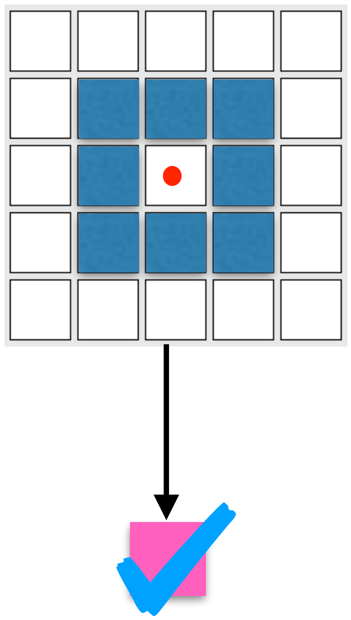


HIDDEN MARKOV MODEL (HMM)

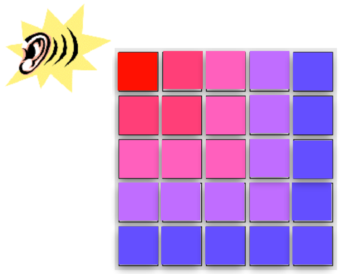


Eg: say observations were

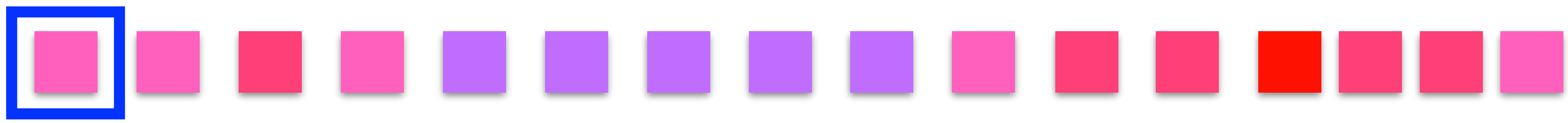


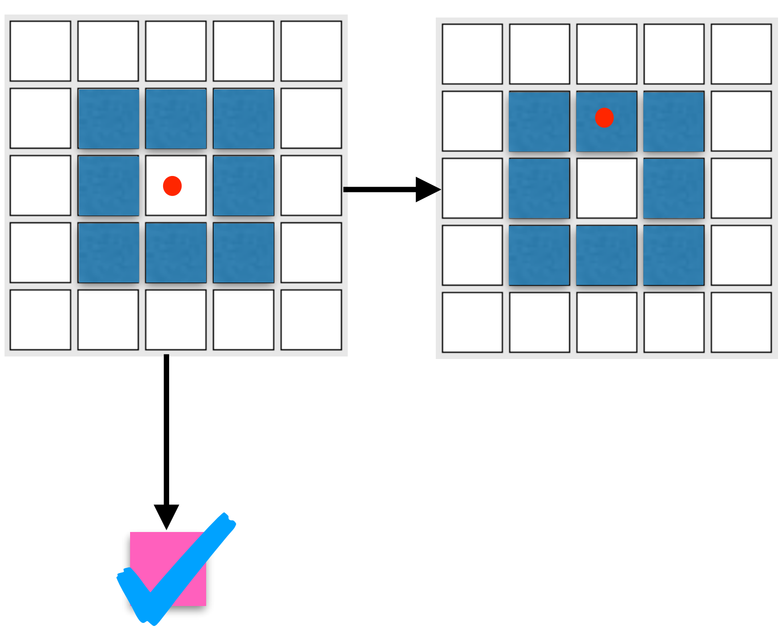


Hidden Markov Model (HMM)

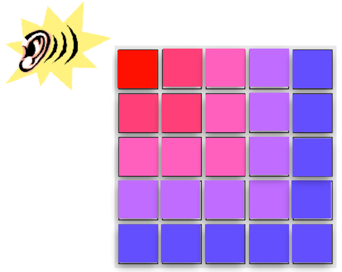


Eg: say observations were

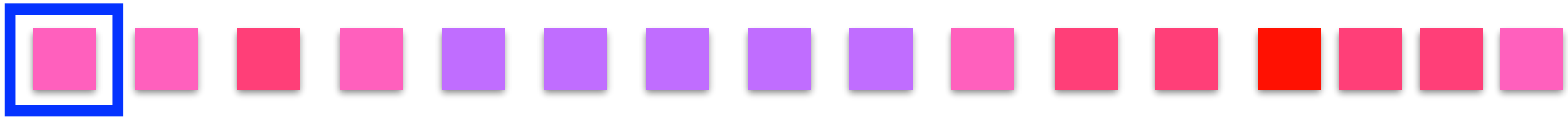


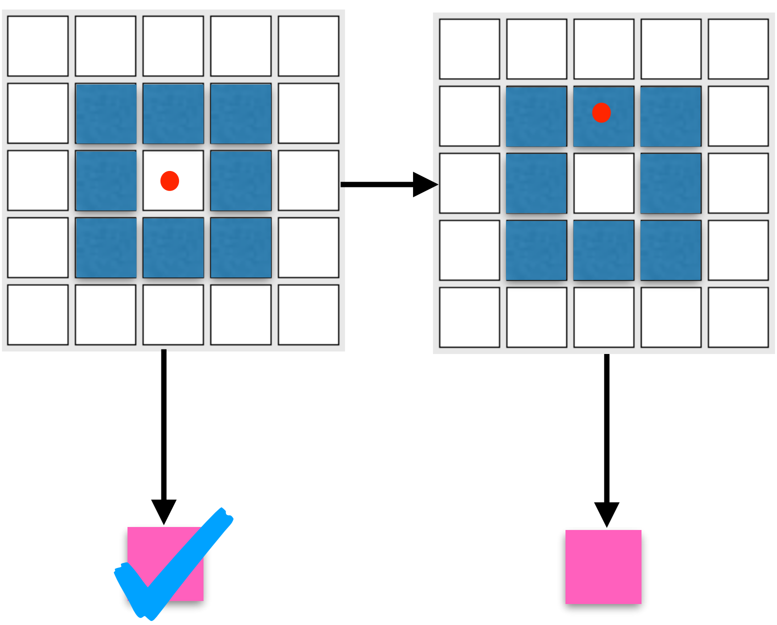


Hidden Markov Model (HMM)

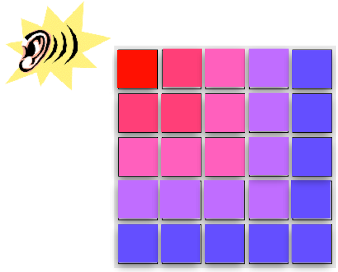


Eg: say observations were

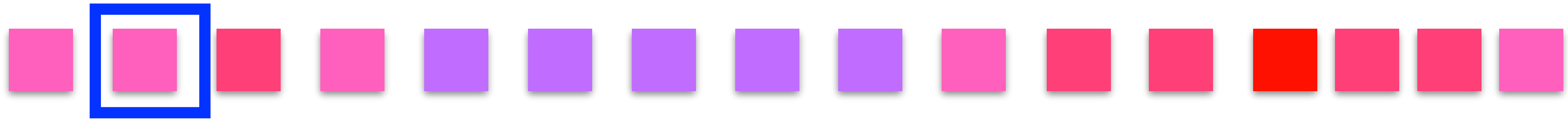


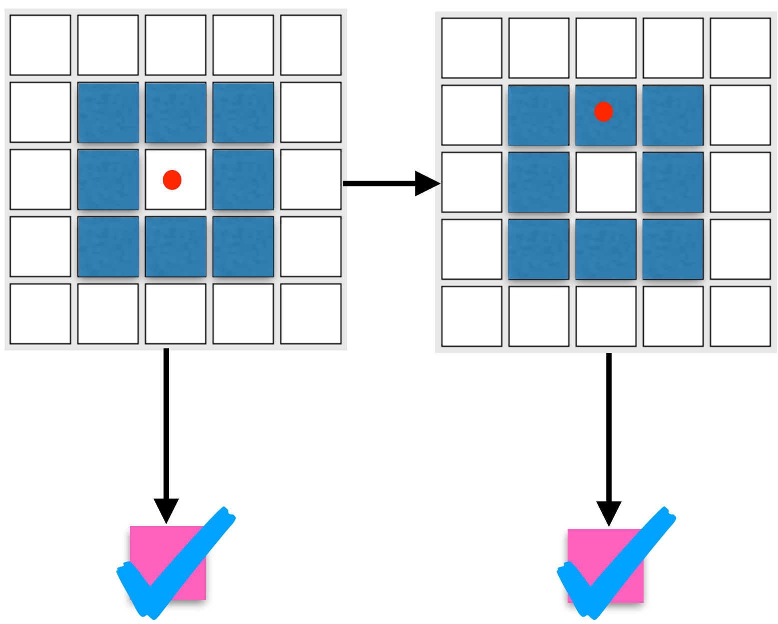


HIDDEN MARKOV MODEL (HMM)

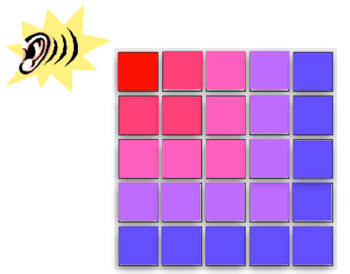


Eg: say observations were

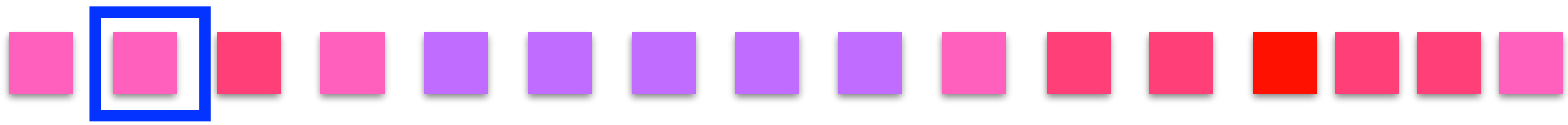


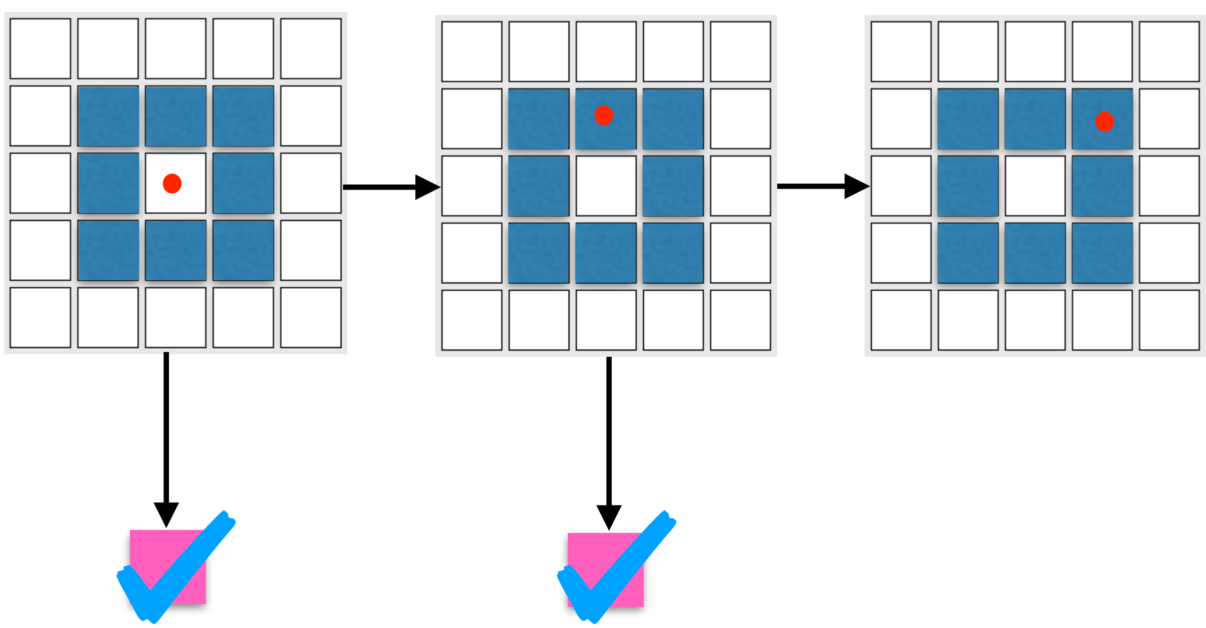


Hidden Markov Model (HMM)

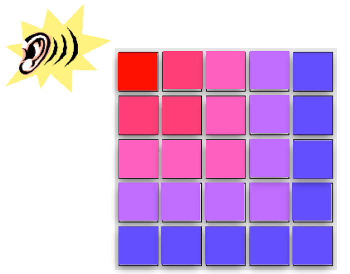


Eg: say observations were

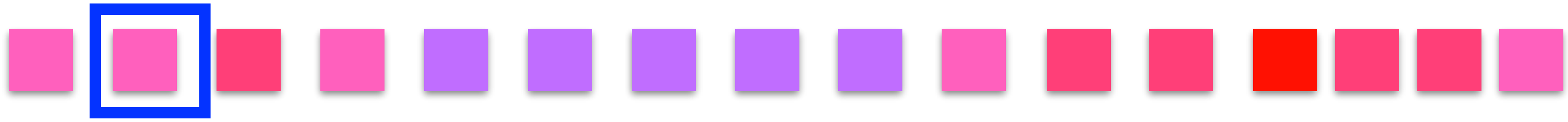


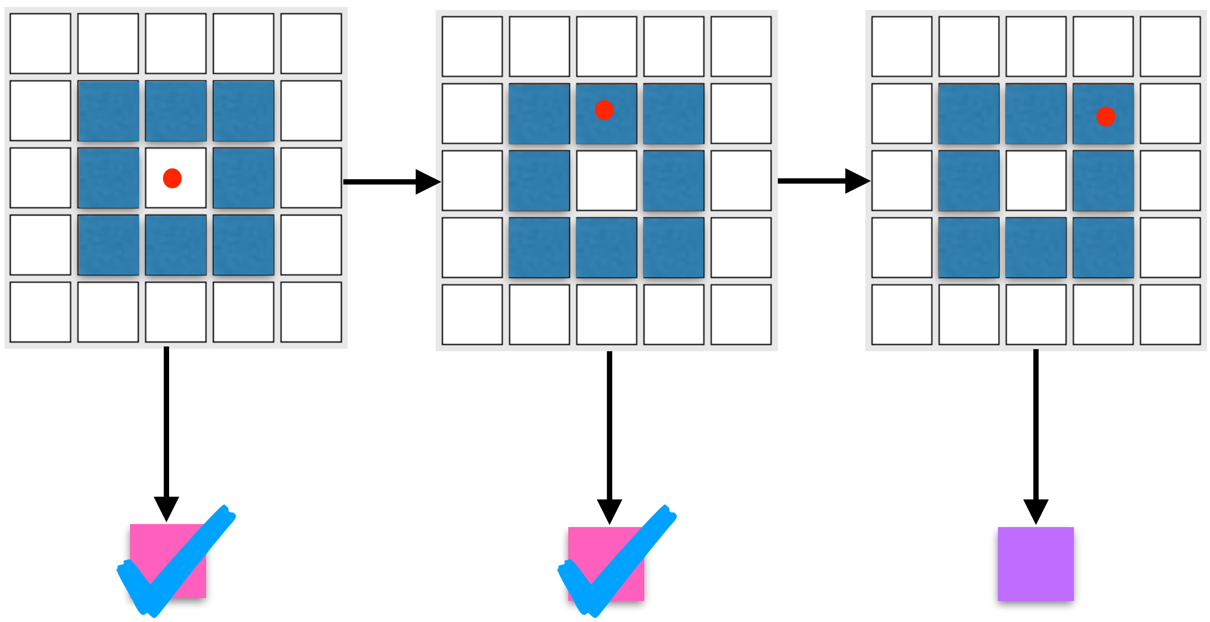


Hidden Markov Model (HMM)



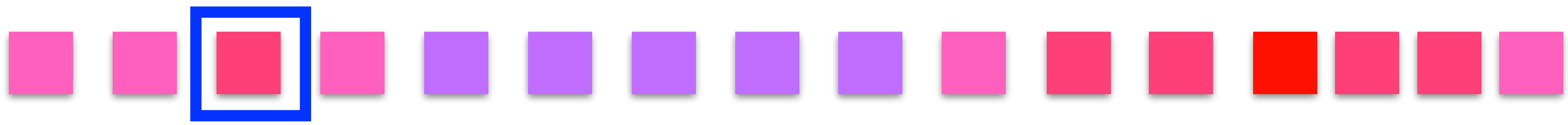
Eg: say observations were

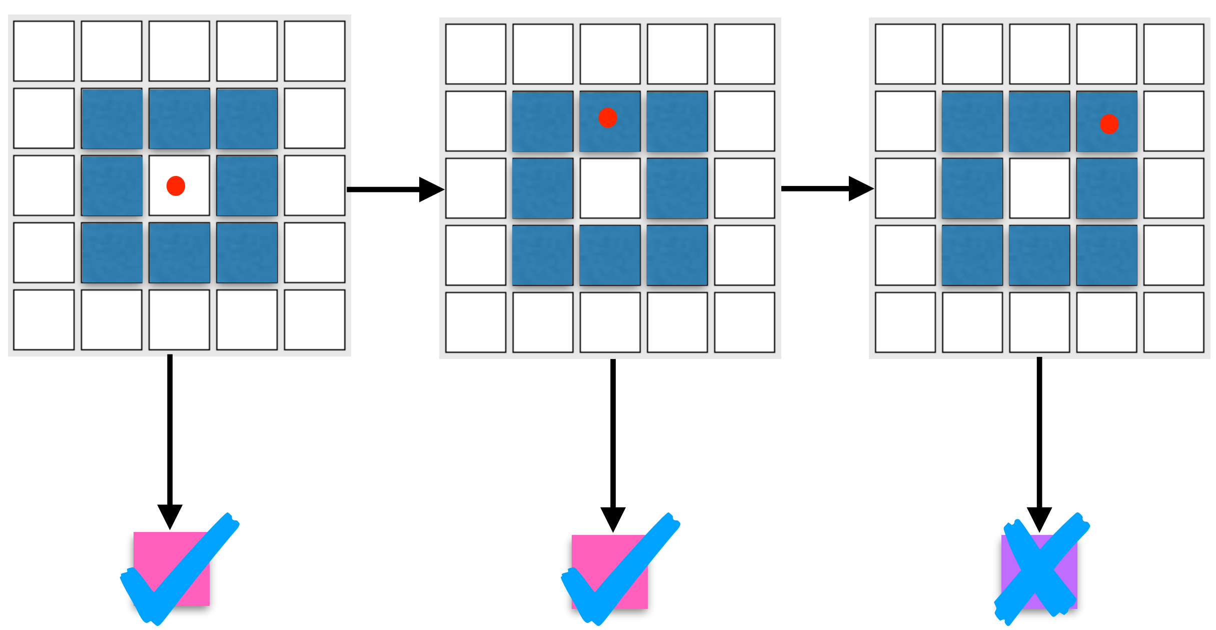


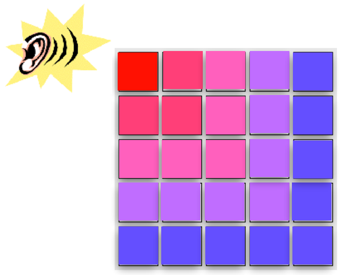


Hidden Markov Model (HMM)

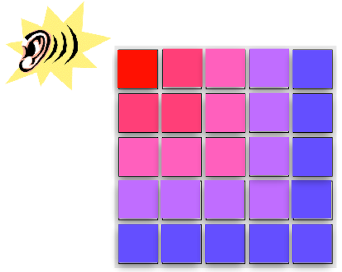
Eg: say observations were



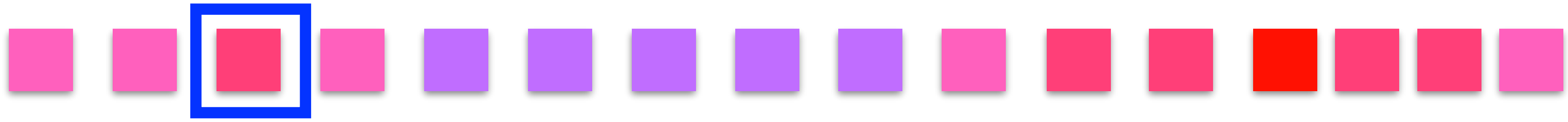


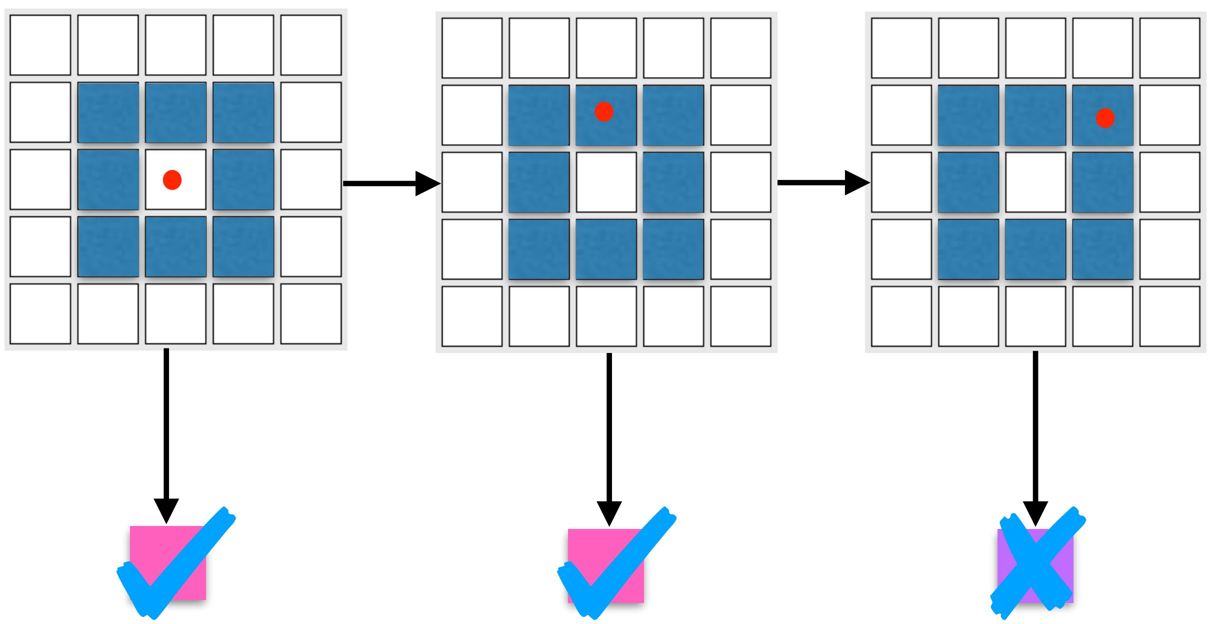


Hidden Markov Model (HMM)



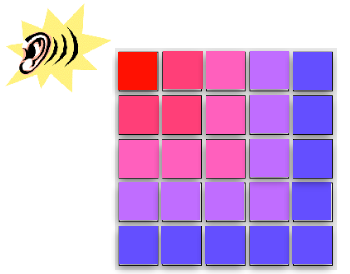
Eg: say observations were



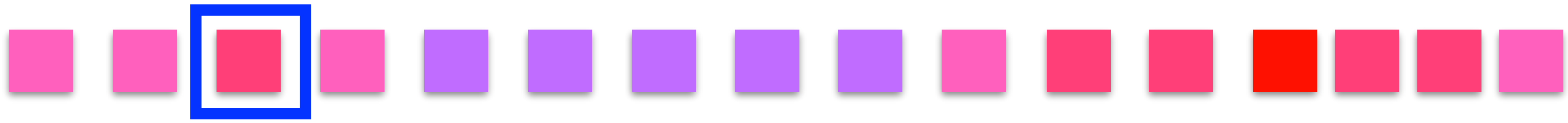


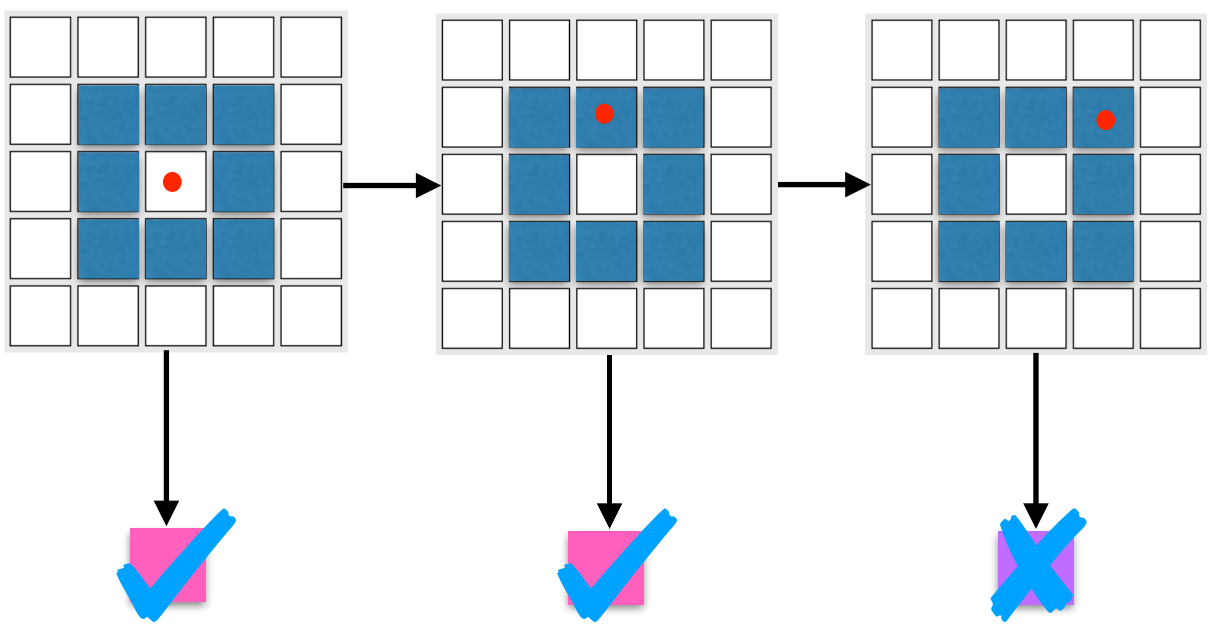
Problem: Most samples rejected

Hidden Markov Model (HMM)



Eg: say observations were

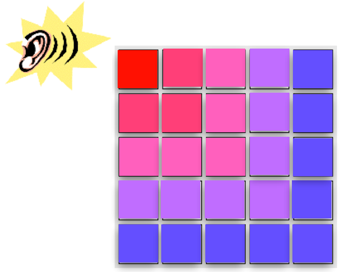




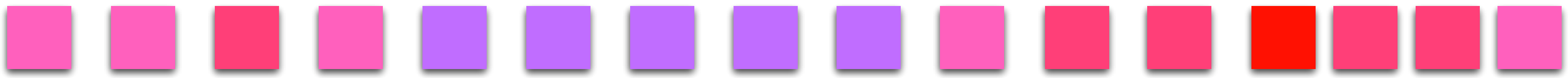
Multiple samples simultaneously.

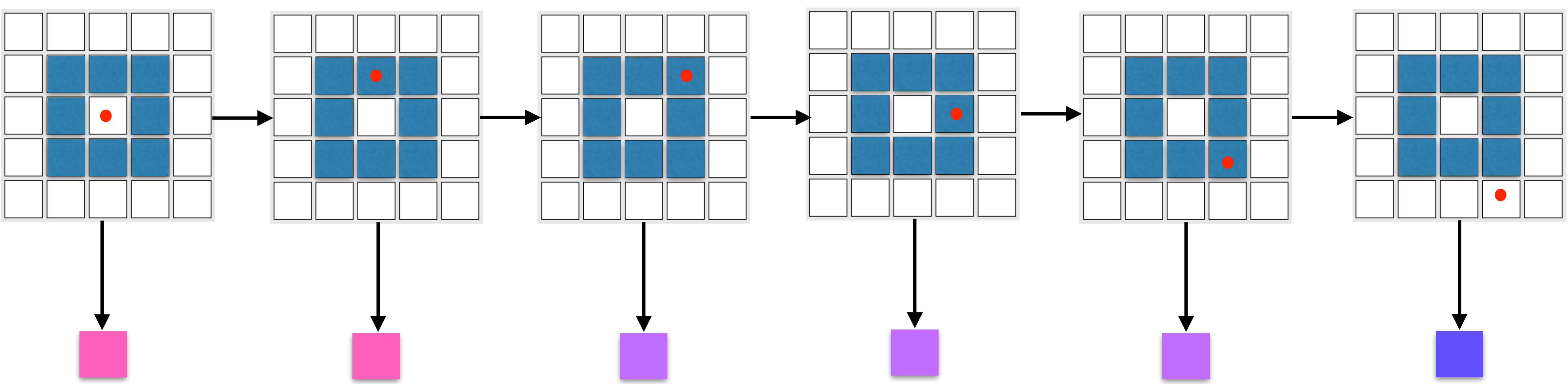
Problem: Most samples rejected

HIDDEN MARKOV MODEL (HMM)

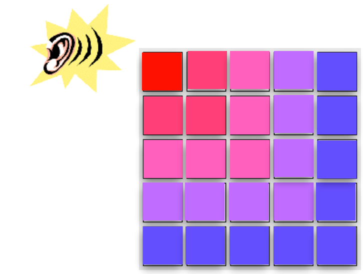


Eg: say observations were

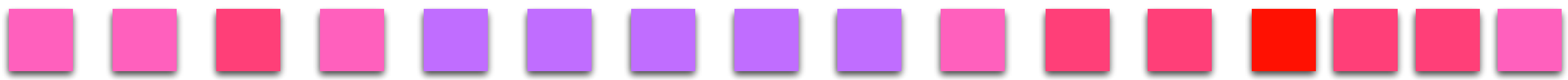


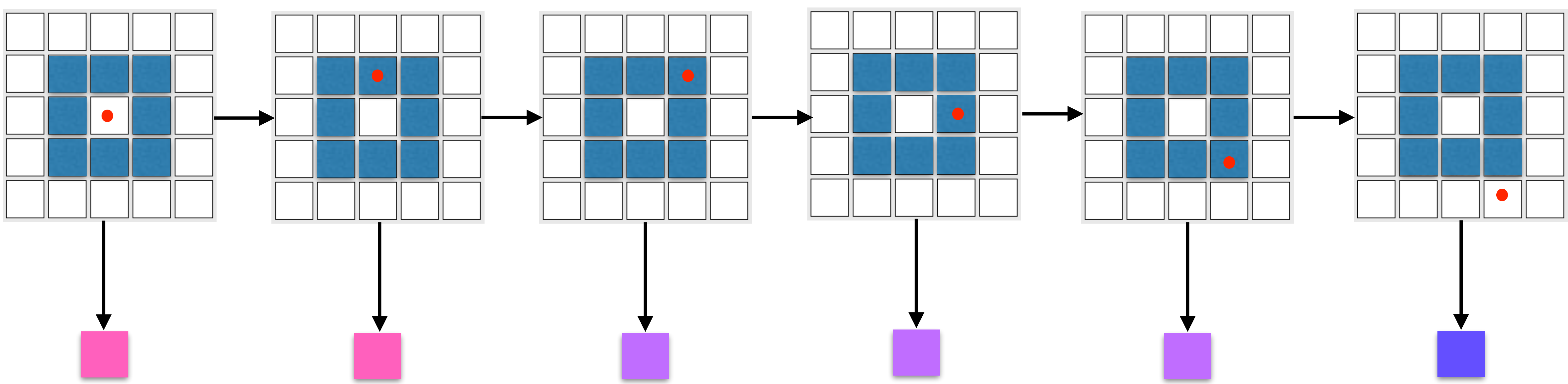


HIDDEN MARKOV MODEL (HMM)



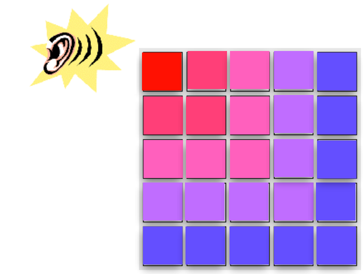
Eg: say observations were



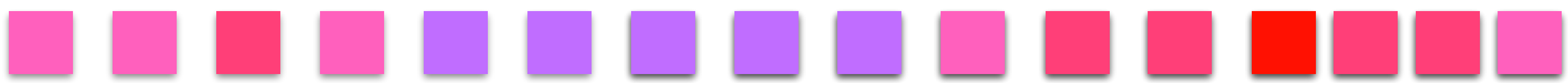


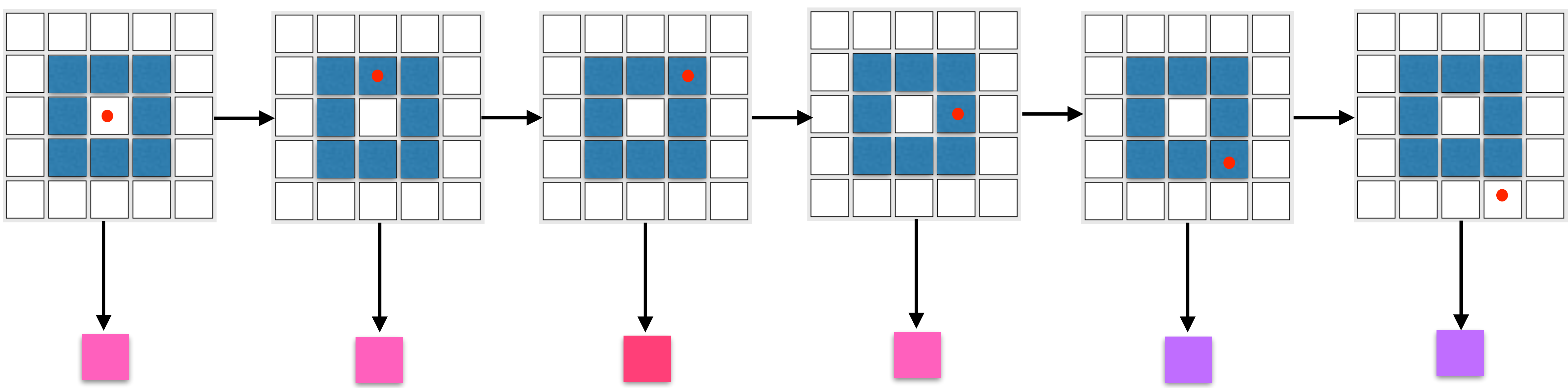
Importance weighting: weight samples

Hidden Markov Model (HMM)



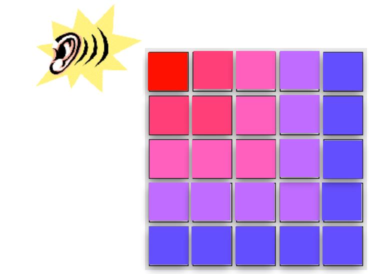
Eg: say observations were



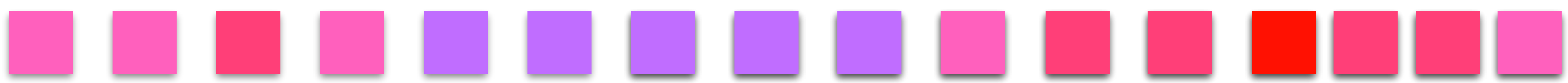


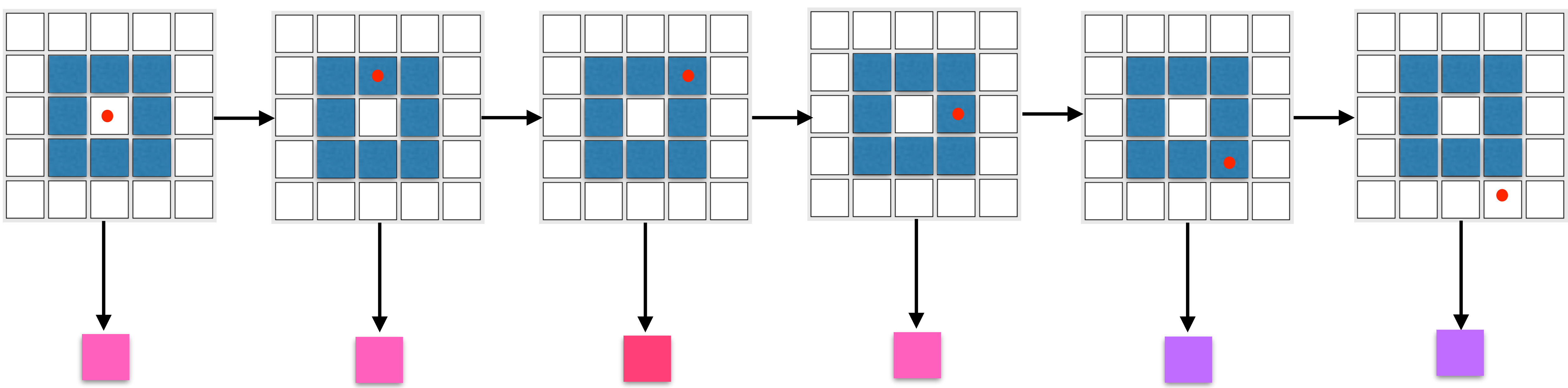
Importance weighting: weight samples

HIDDEN MARKOV MODEL (HMM)



Eg: say observations were

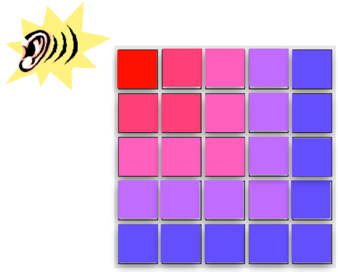




$P(\;\;| X_1=13) \times P(\;\;| X_2=8) \times P(\;\;| X_3=9) \times P(\;\;| X_5=24) \times P(\;\;| X_5=19) \times P(\;\;| X_4=14)$

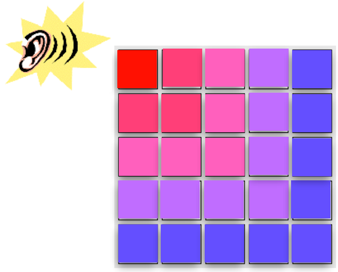
Importance weighting: weight samples

HMM Particle Filter

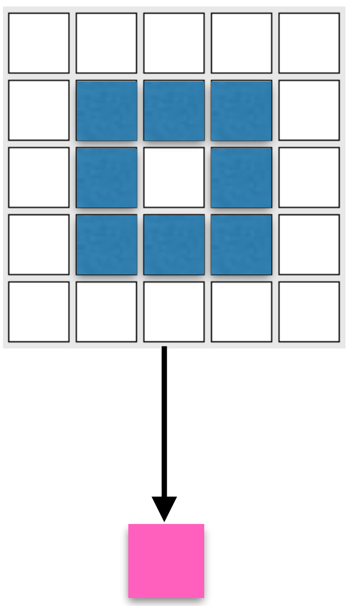


- Use multiple samples and track each ones weights.

HMM Particle Filter

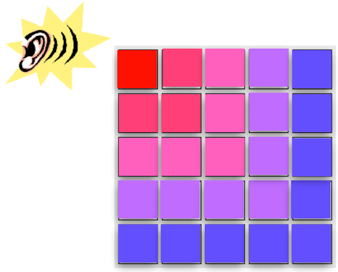


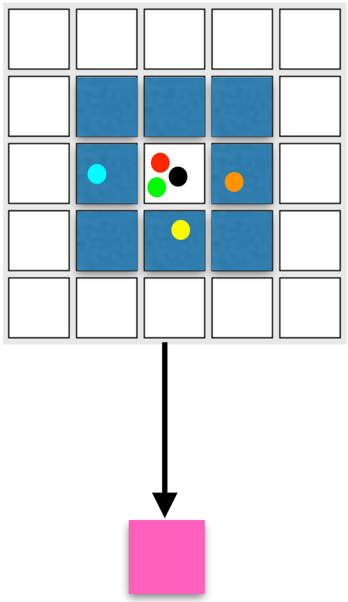
- Use multiple samples and track each ones weights.



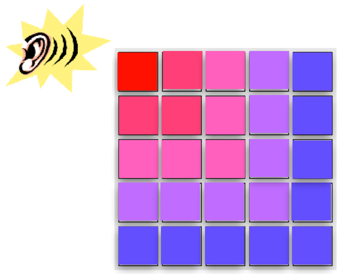
HMM Particle Filter

- Use multiple samples and track each ones weights.

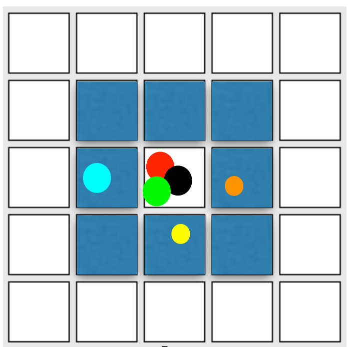


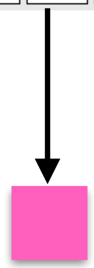


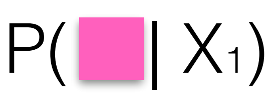
HMM Particle Filter



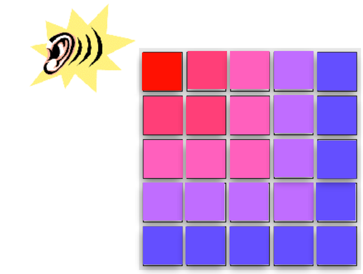
- Use multiple samples and track each ones weights.



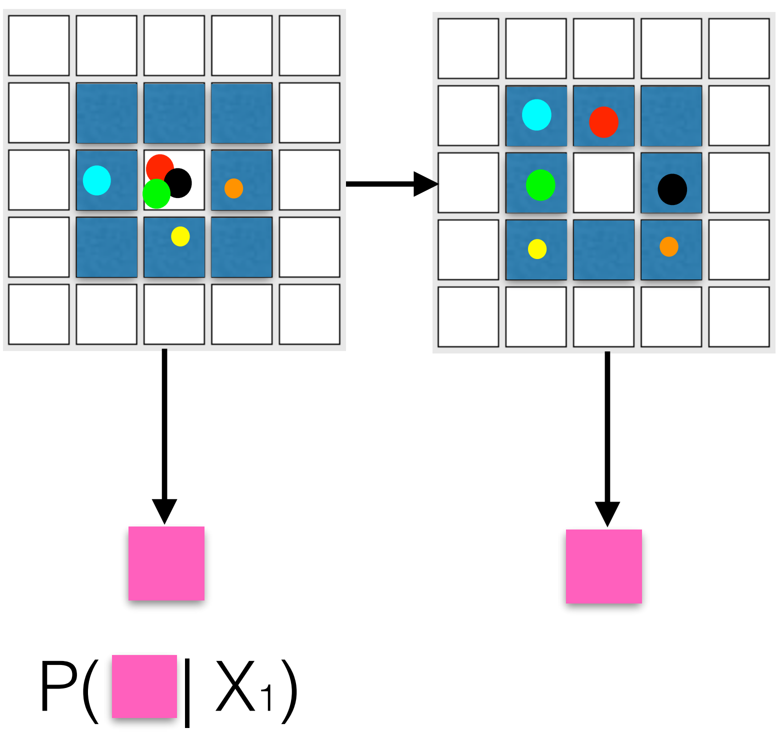




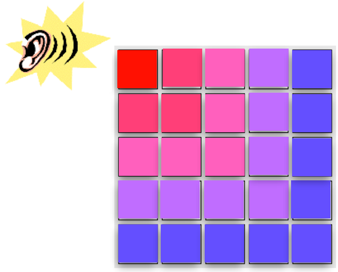
HMM Particle Filter



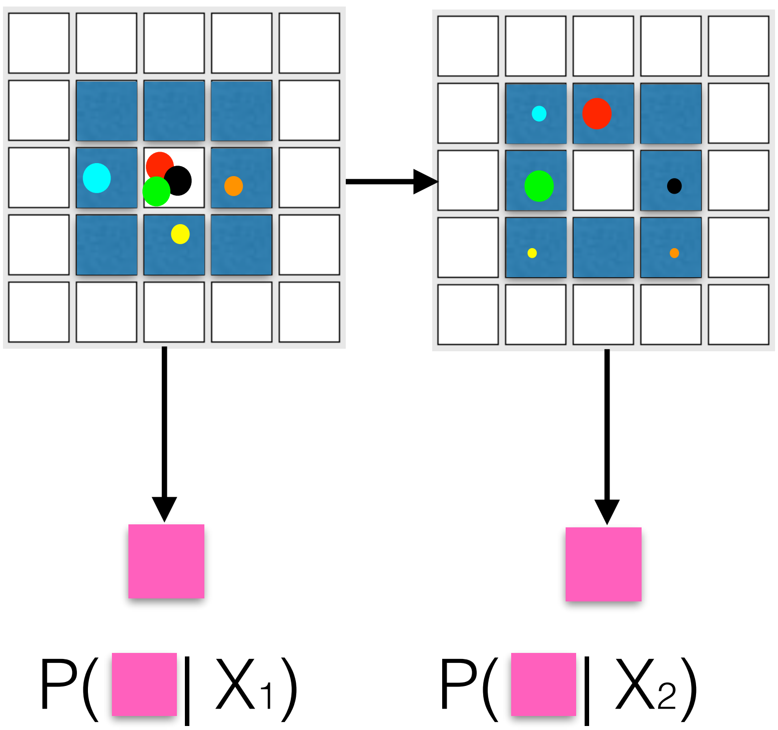
- Use multiple samples and track each ones weights.



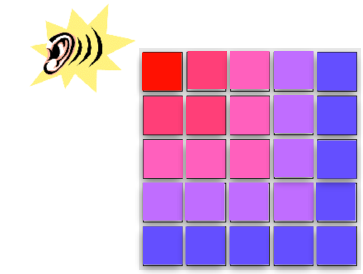
HMM Particle Filter



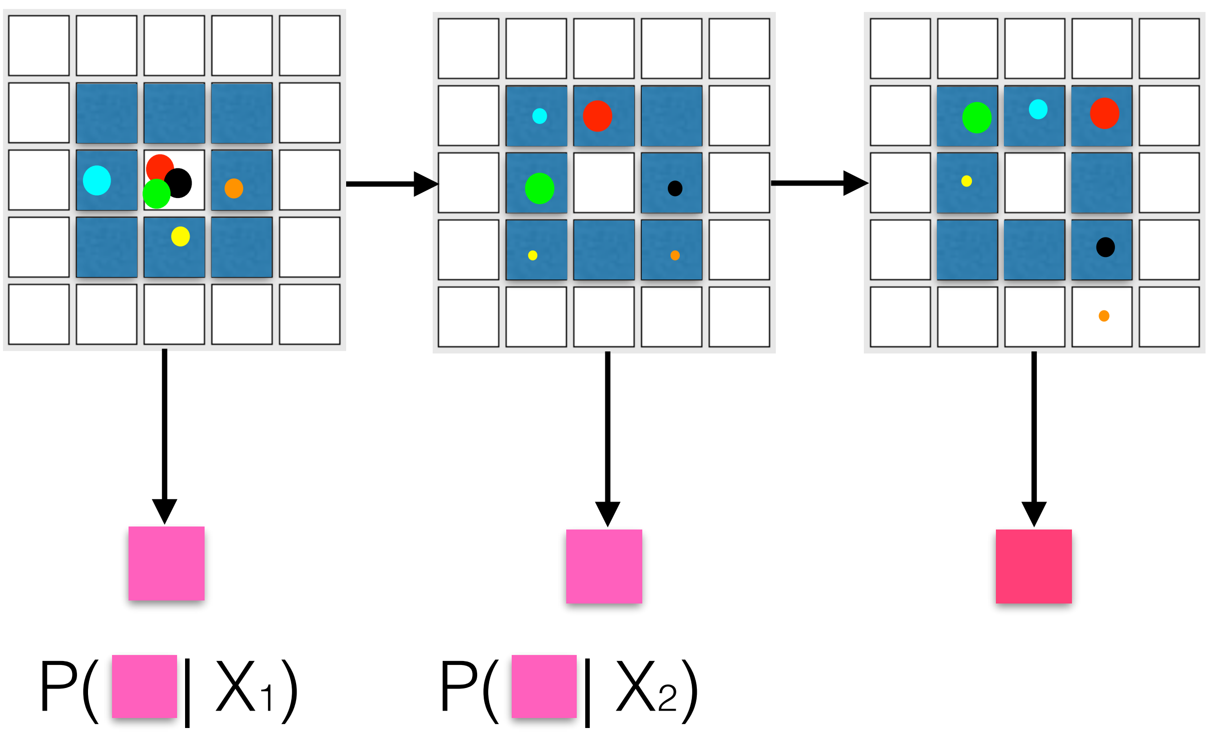
- Use multiple samples and track each ones weights.



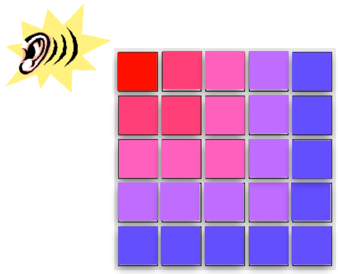
HMM Particle Filter



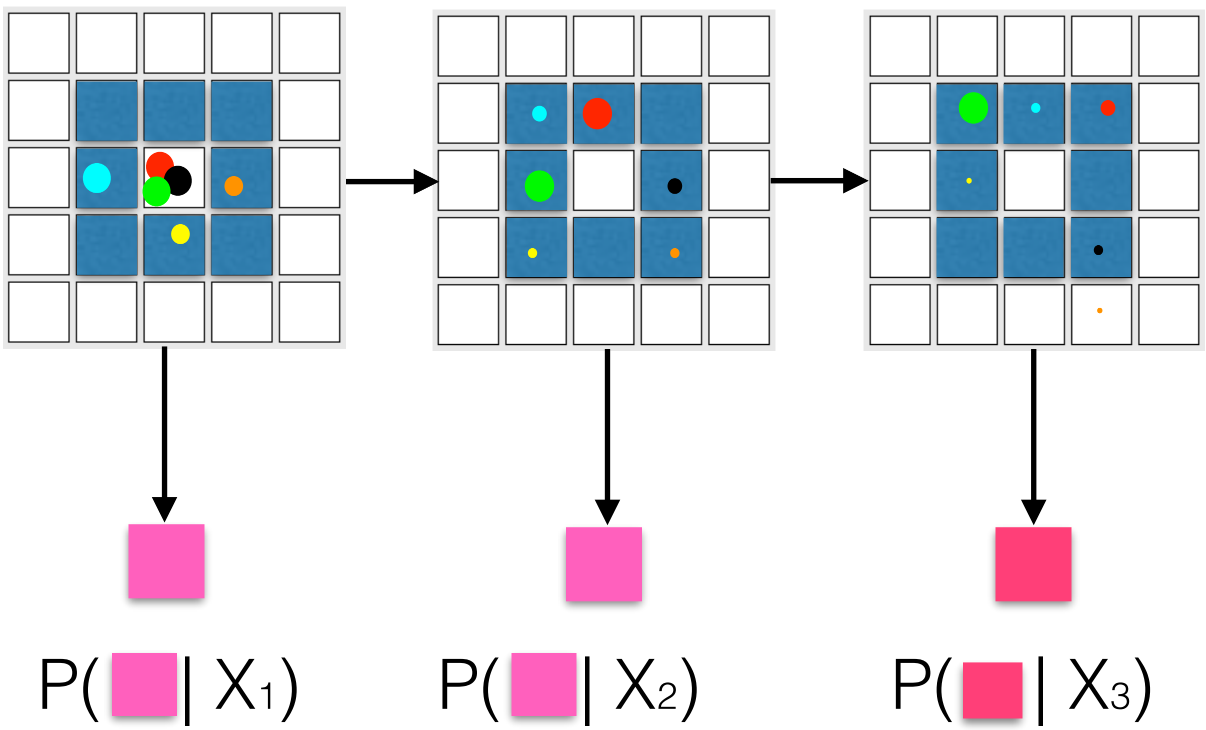
- Use multiple samples and track each ones weights.



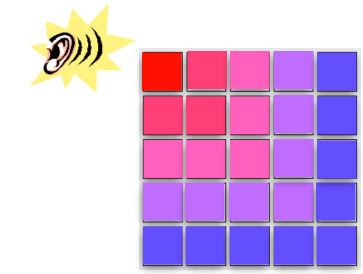
HMM Particle Filter



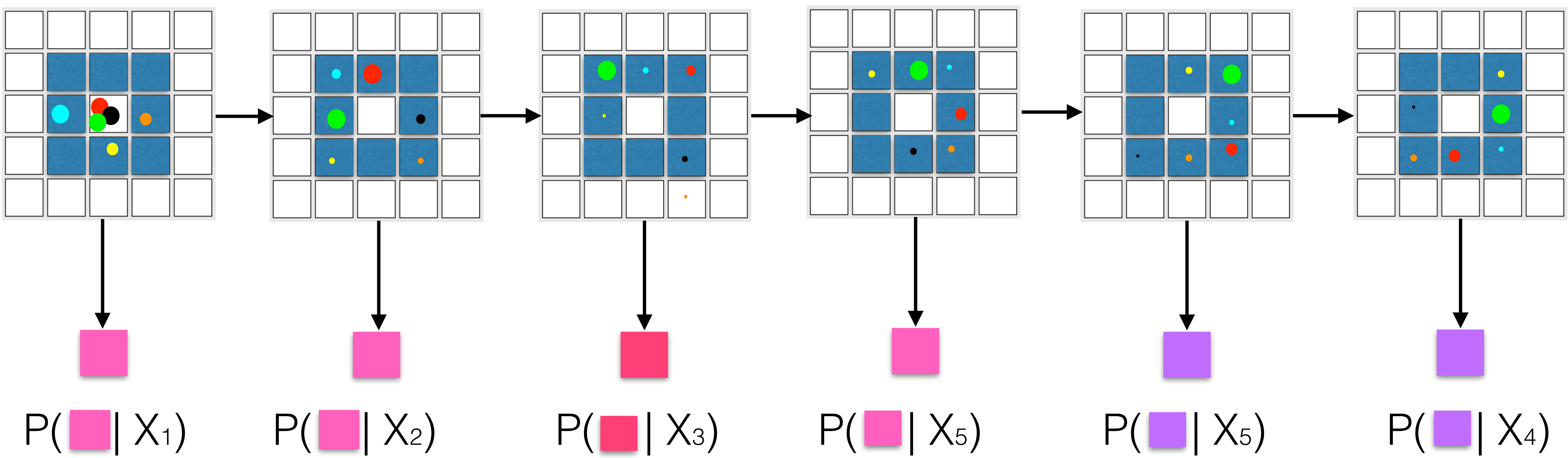
- Use multiple samples and track each ones weights.



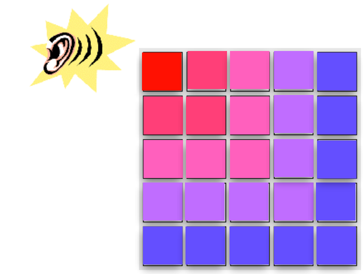
HMM Particle Filter



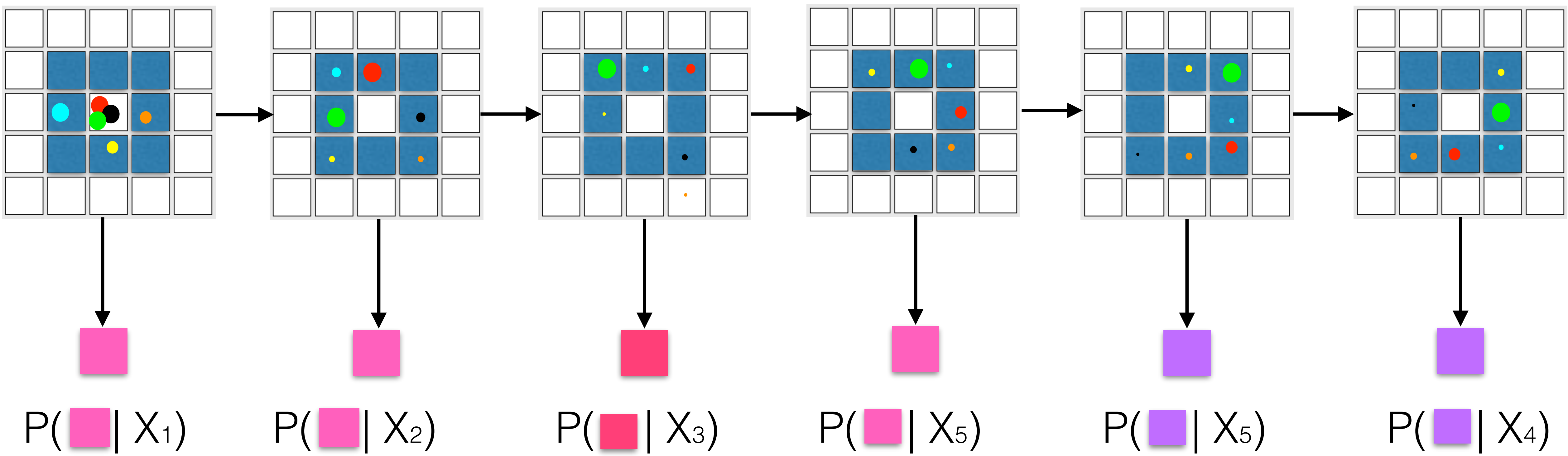
- Use multiple samples and track each ones weights.



HMM Particle Filter

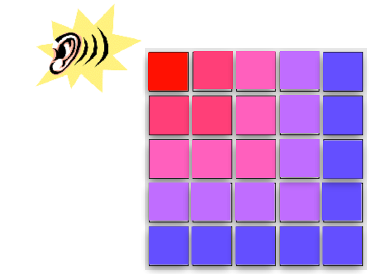


- Use multiple samples and track each ones weights.

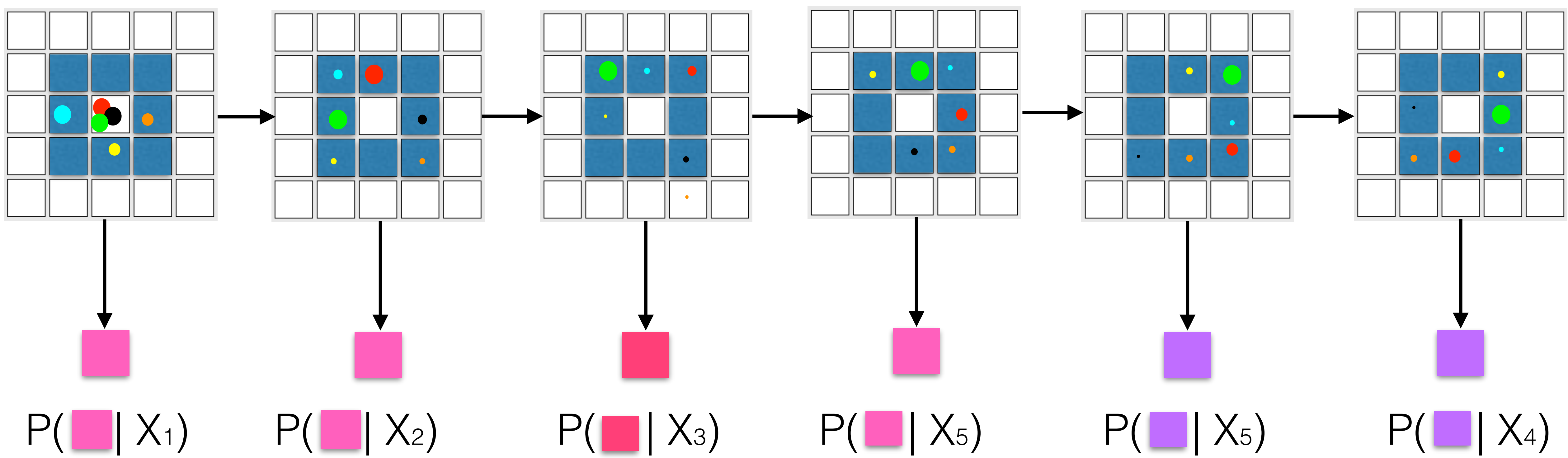


- This is same as 6 separate samples

HMM PARTICLE FILTER

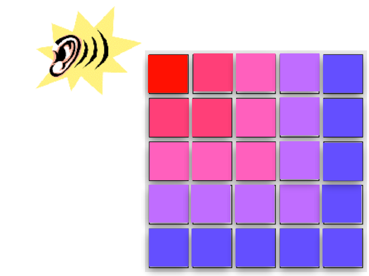


- Use multiple samples and track each ones weights.

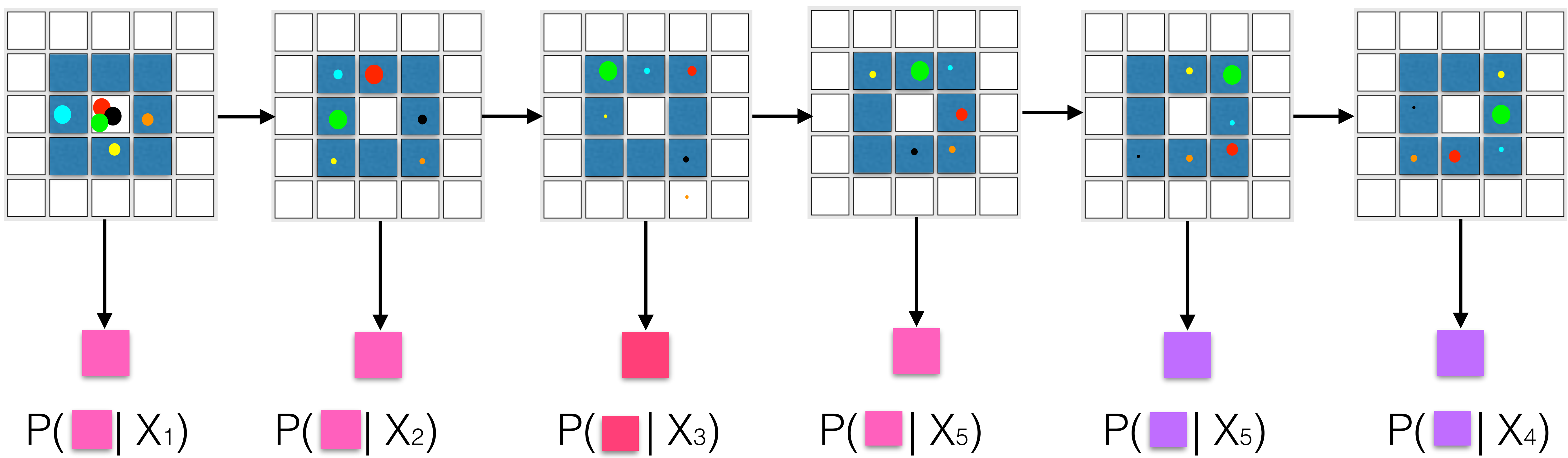


- This is same as 6 separate samples
- Instead of tracking each sample’s weight, resample according to weights

HMM Particle Filter

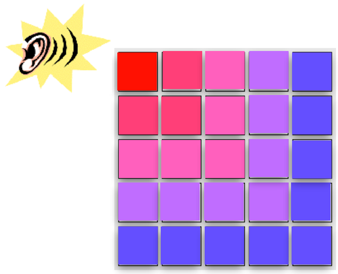


- Use multiple samples and track each ones weights.

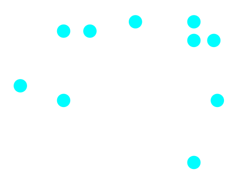


- This is same as 6 separate samples
- Instead of tracking each sample's weight, resample according to weights
- Problem: Too many samples have negligible weight!

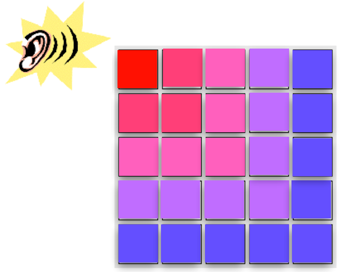
HMM Particle Filter



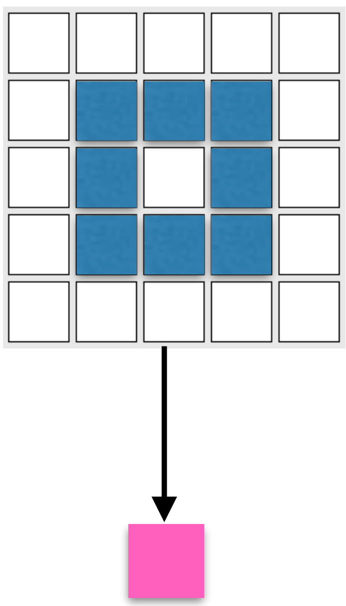
Instead of tracking each one, resample!



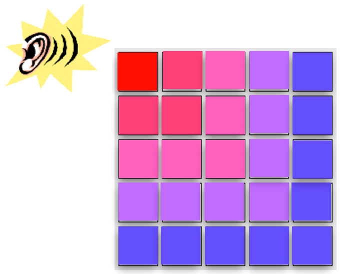
HMM Particle Filter



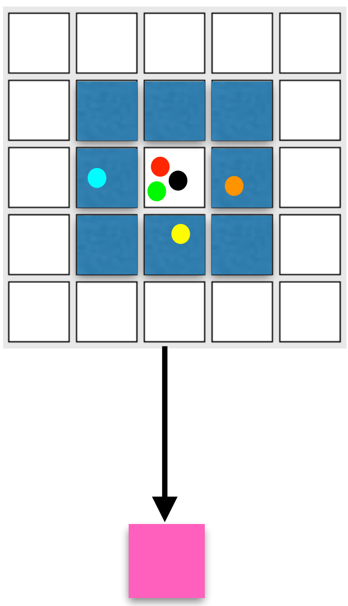
Instead of tracking each one, resample!



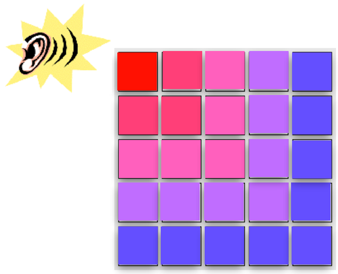
HMM Particle Filter



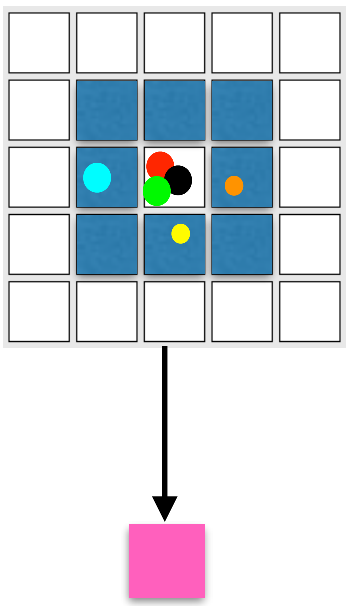
Instead of tracking each one, resample!

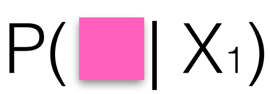


HMM Particle Filter

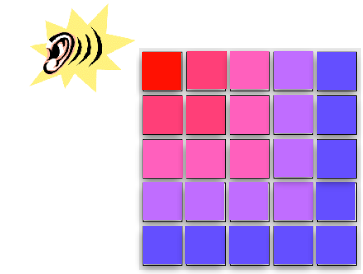


Instead of tracking each one, resample!

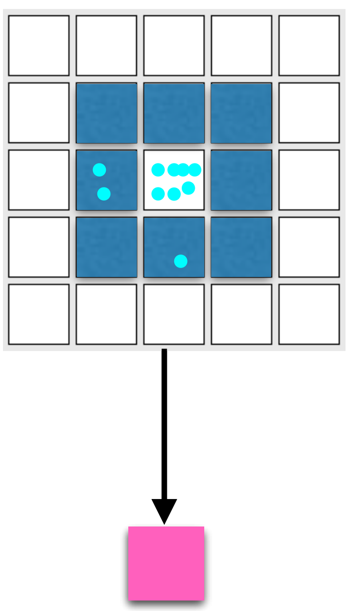


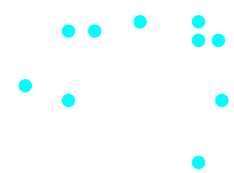


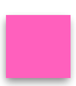
HMM Particle Filter



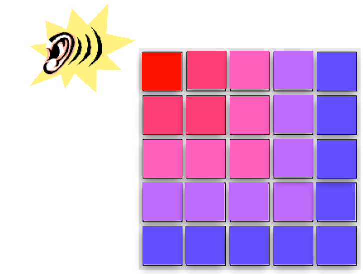
Instead of tracking each one, resample!



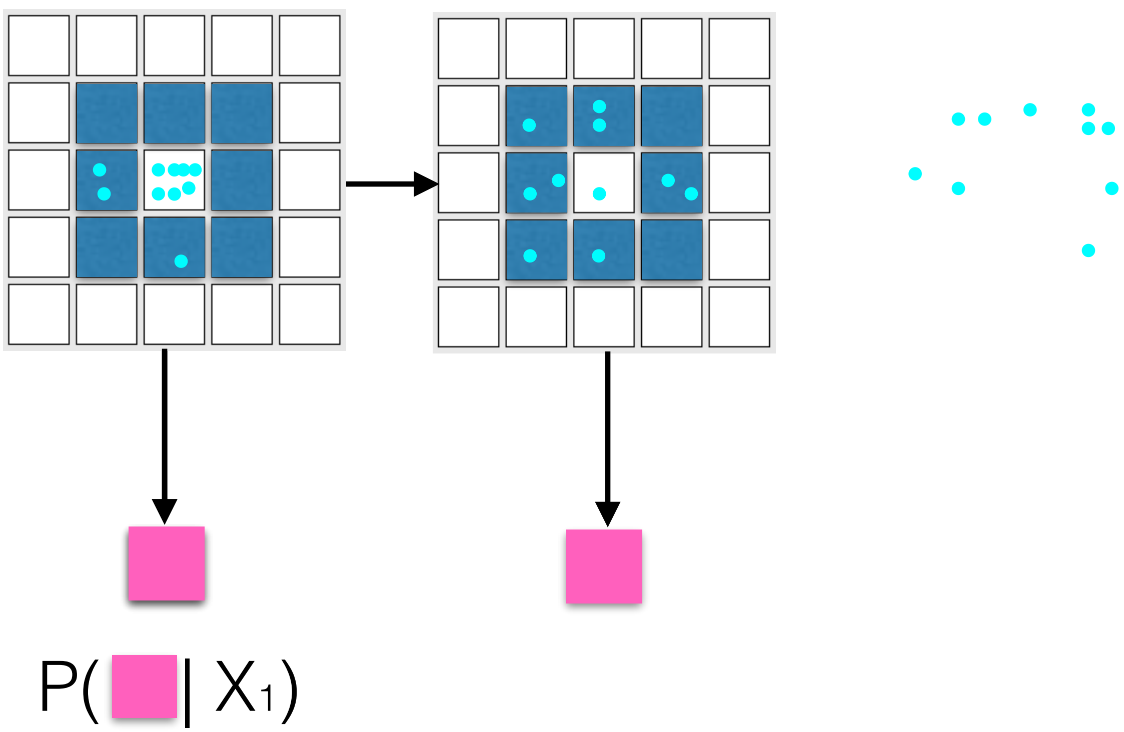


P( | X_1)

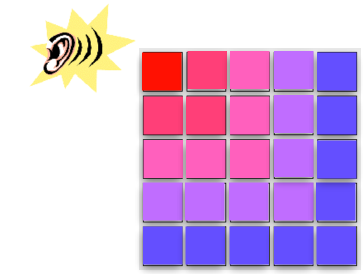
HMM Particle Filter



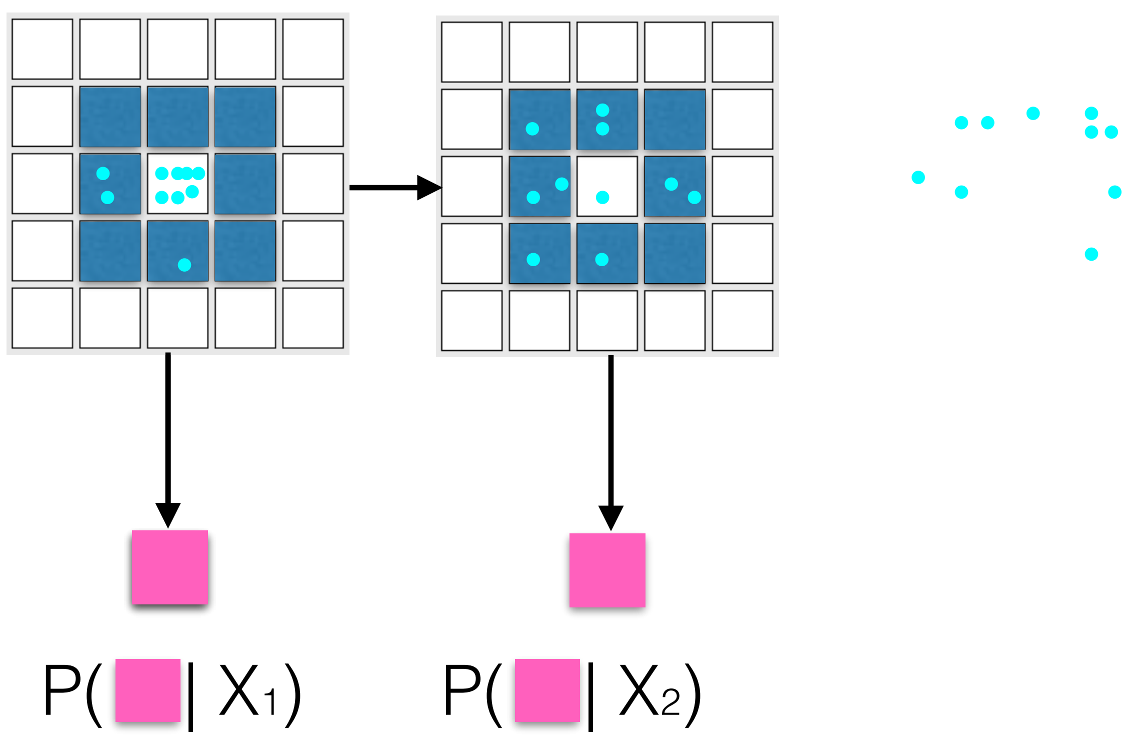
Instead of tracking each one, resample!



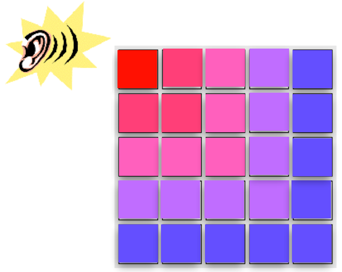
HMM Particle Filter



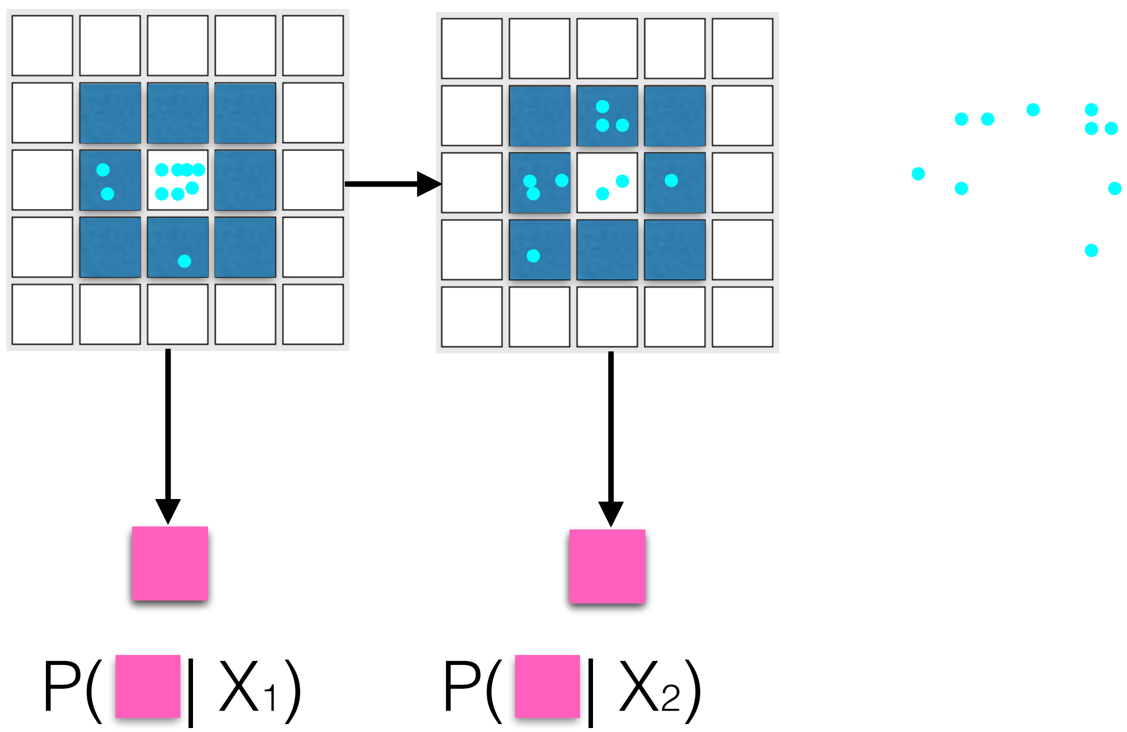
Instead of tracking each one, resample!



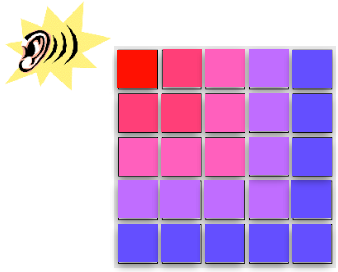
HMM Particle Filter



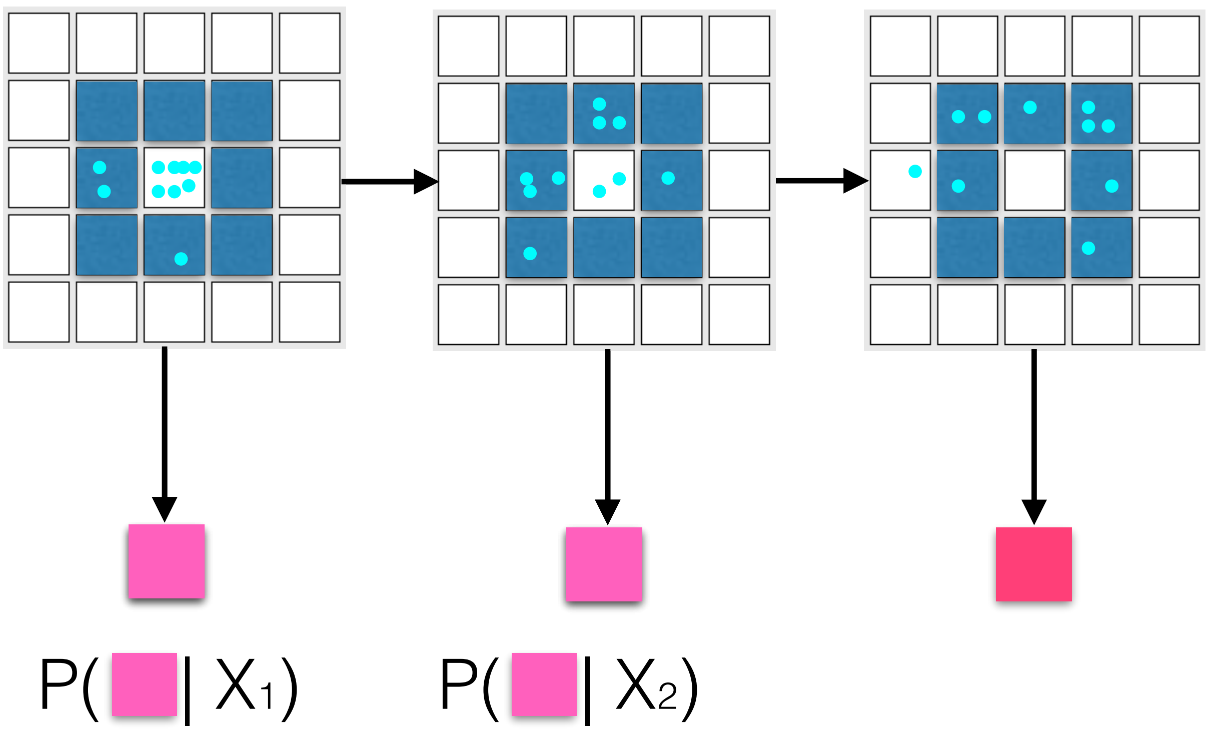
Instead of tracking each one, resample!



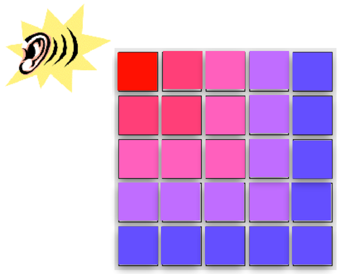
HMM Particle Filter



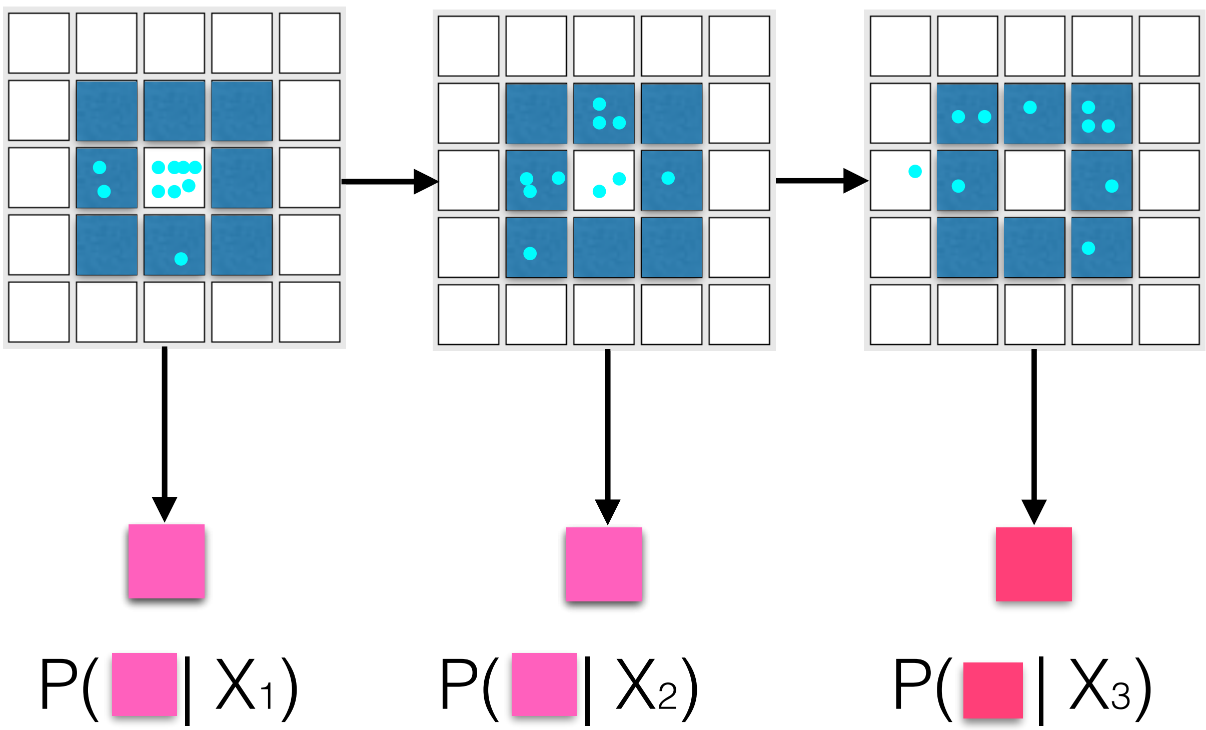
Instead of tracking each one, resample!



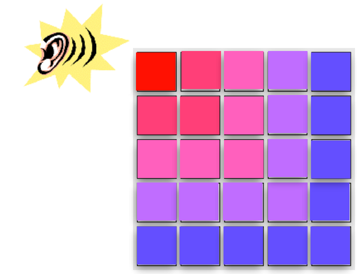
HMM Particle Filter



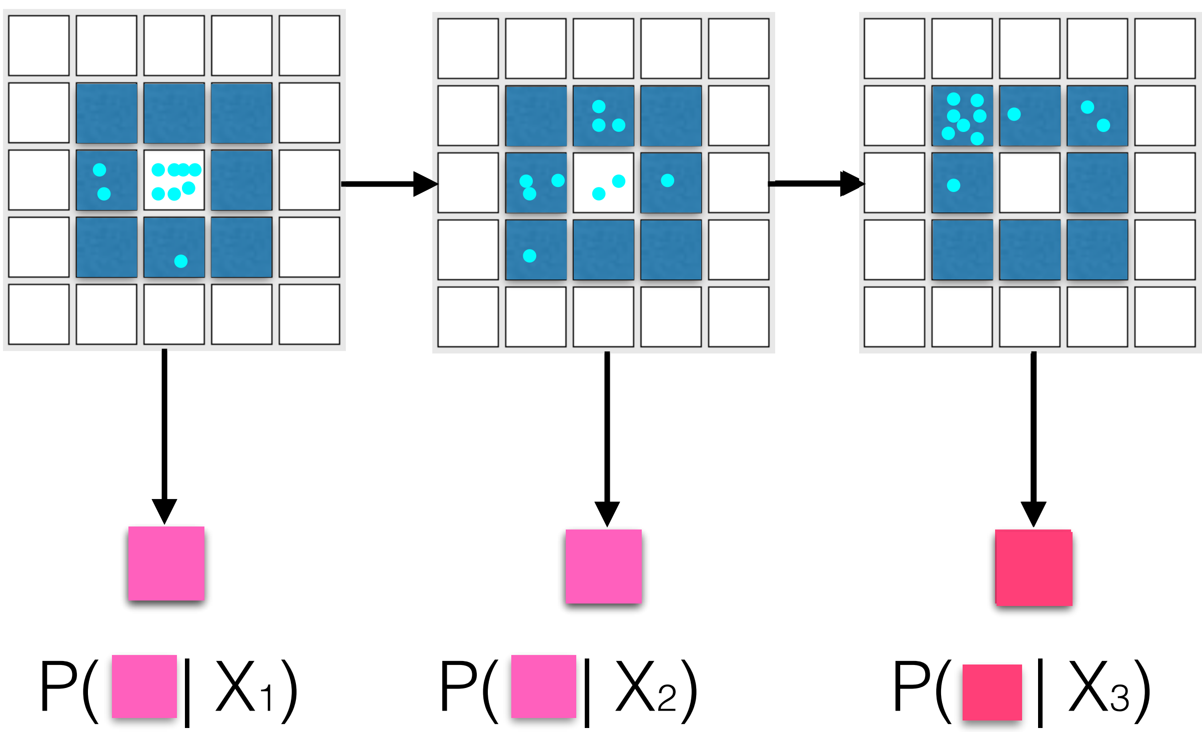
Instead of tracking each one, resample!



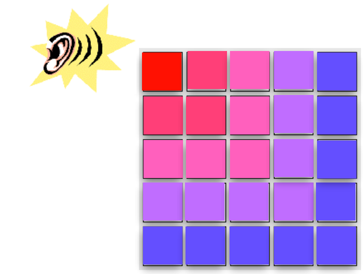
HMM Particle Filter



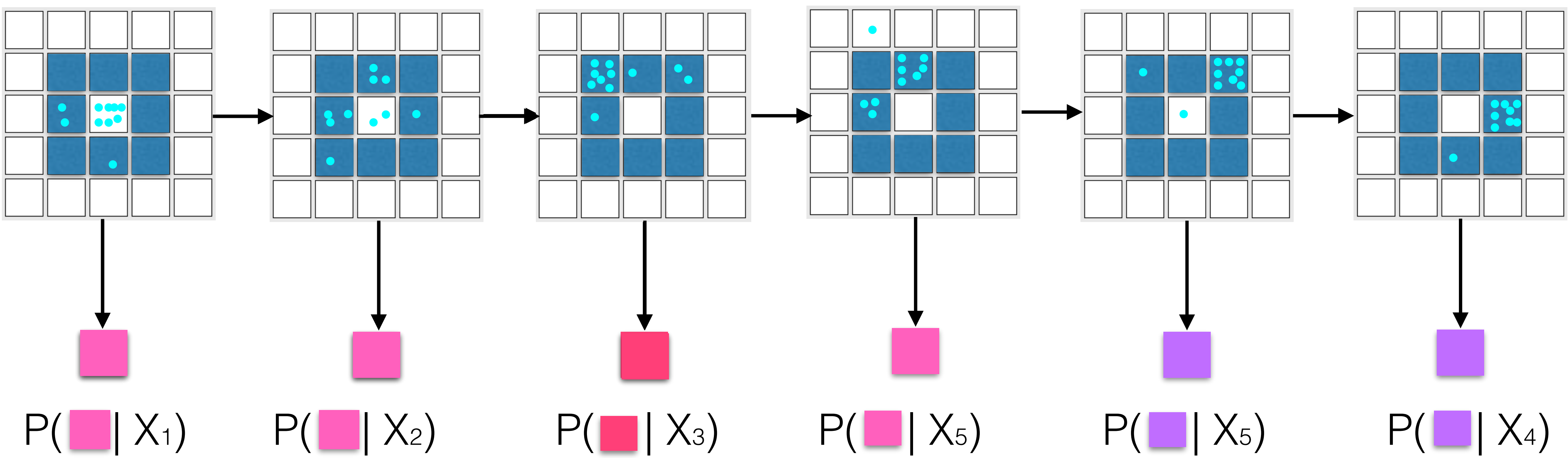
Instead of tracking each one, resample!



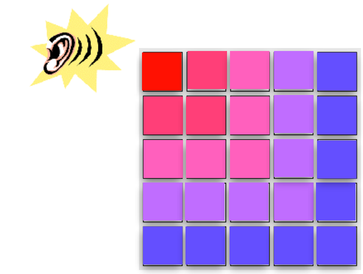
HMM Particle Filter



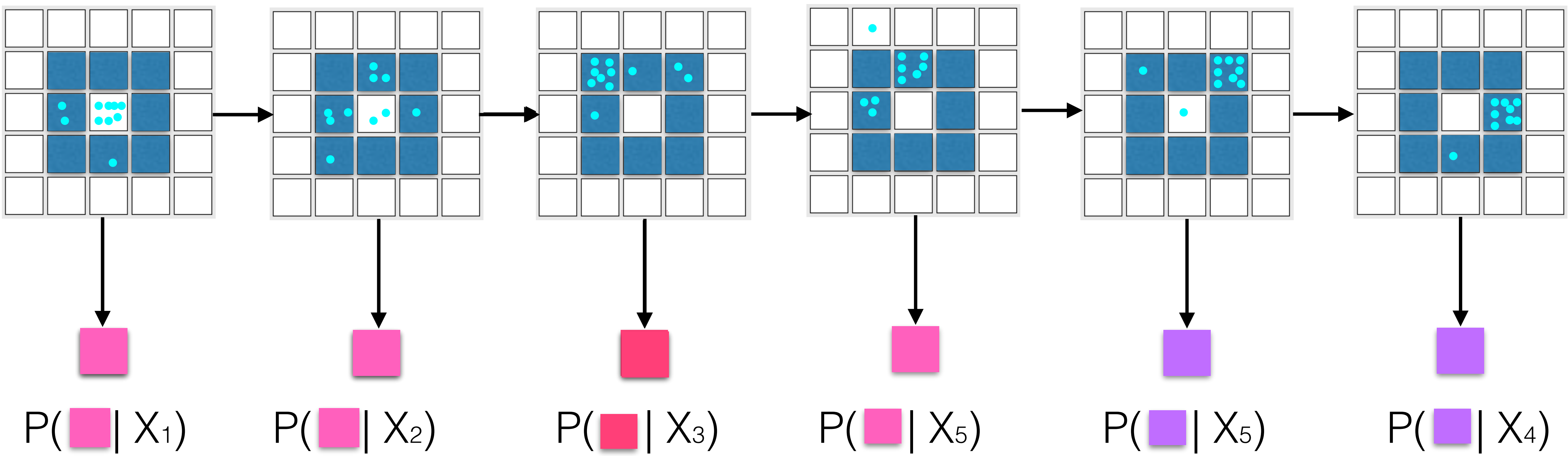
Instead of tracking each one, resample!



HMM Particle Filter

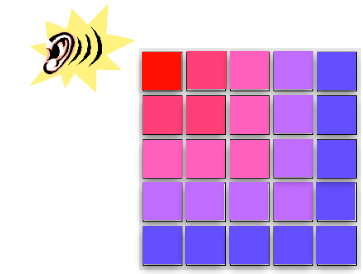


Instead of tracking each one, resample!

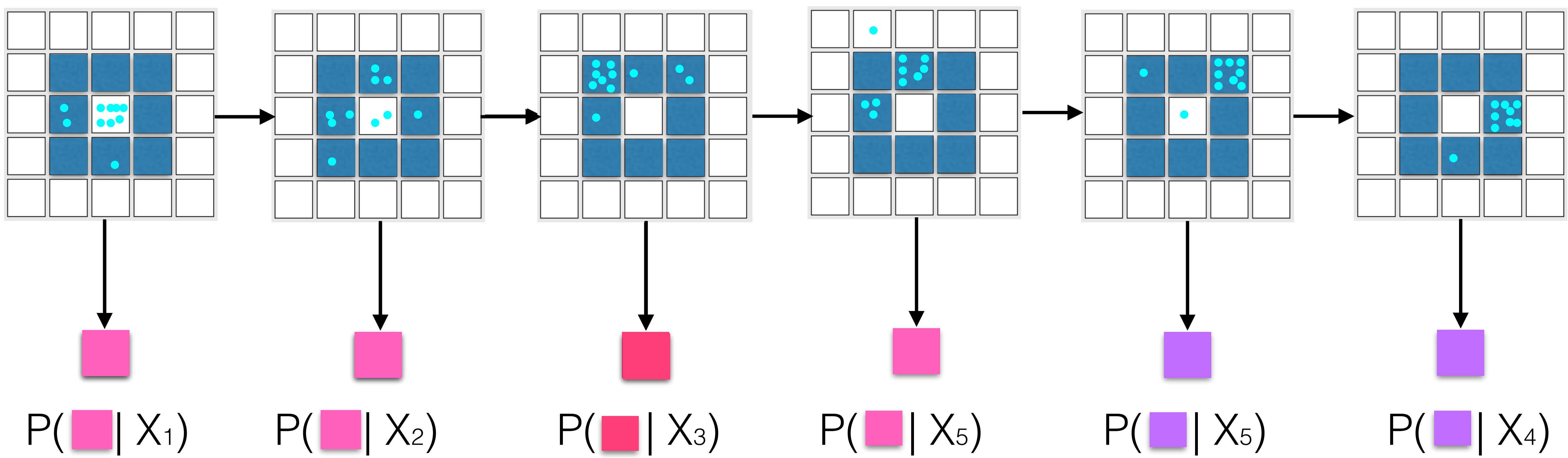


- On every round, transfer particles from previous states according to transition probability

HMM Particle Filter

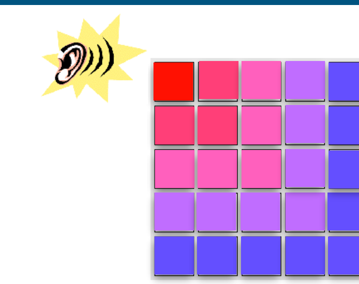


Instead of tracking each one, resample!

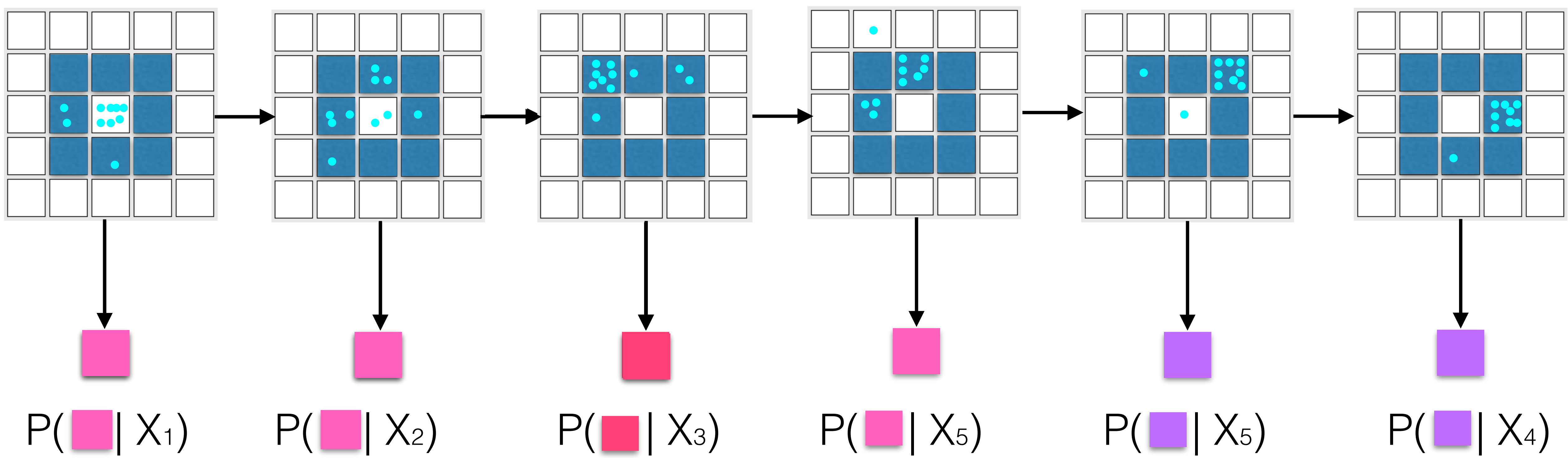


- On every round, transfer particles from previous states according to transition probability
- Resample particles according to P(observation|state)

HMM Particle Filter



Instead of tracking each one, resample!



- On every round, transfer particles from previous states according to transition probability
- Resample particles according to P(observation|state)
- Use new particles to proceed

Particle Filtering

Particle Filtering

- Without resampling, we carry many particles with very small probabilities

Particle Filtering

- Without resampling, we carry many particles with very small probabilities
  - too many samples needed for a good estimate

Particle Filtering

- Without resampling, we carry many particles with very small probabilities
    - too many samples needed for a good estimate
- By resampling, we got rid of samples with very small probabilities

Particle Filtering

- Without resampling, we carry many particles with very small probabilities
  - too many samples needed for a good estimate
- By resampling, we got rid of samples with very small probabilities
  - Hence fewer samples suffice

HMM Particle Filter

- Inference time only depends on number of samples
- Of course more the samples the better accuracy
- Often we don’t need too many samples. Why ?

Gibbs Sampling

- Repeat n times for, n samples,
  - Start with arbitrary value for variables
  - Replace each variable by new sample from P(Variable| all other variables)
  - Go over all variables multiple times
  - Return final sample of the N variables

Variational Inference

- Basic idea: we want to infer $P(\text{Unobserved}|\text{Observed})$
  We create a new parametric distribution $Q_\theta(\text{Unobserved})$ where
  θ is picked based on Obervations
- We pick θ such that, Q_θ is close to $P(\text{Unobserved}|\text{Observed})$
- Closeness measured using KL divergence
- Mean-field approximation,

$$Q_\theta(X_1, \ldots, X_m) = \prod_{j=1}^{m} Q_{\theta_j}(X_j)$$