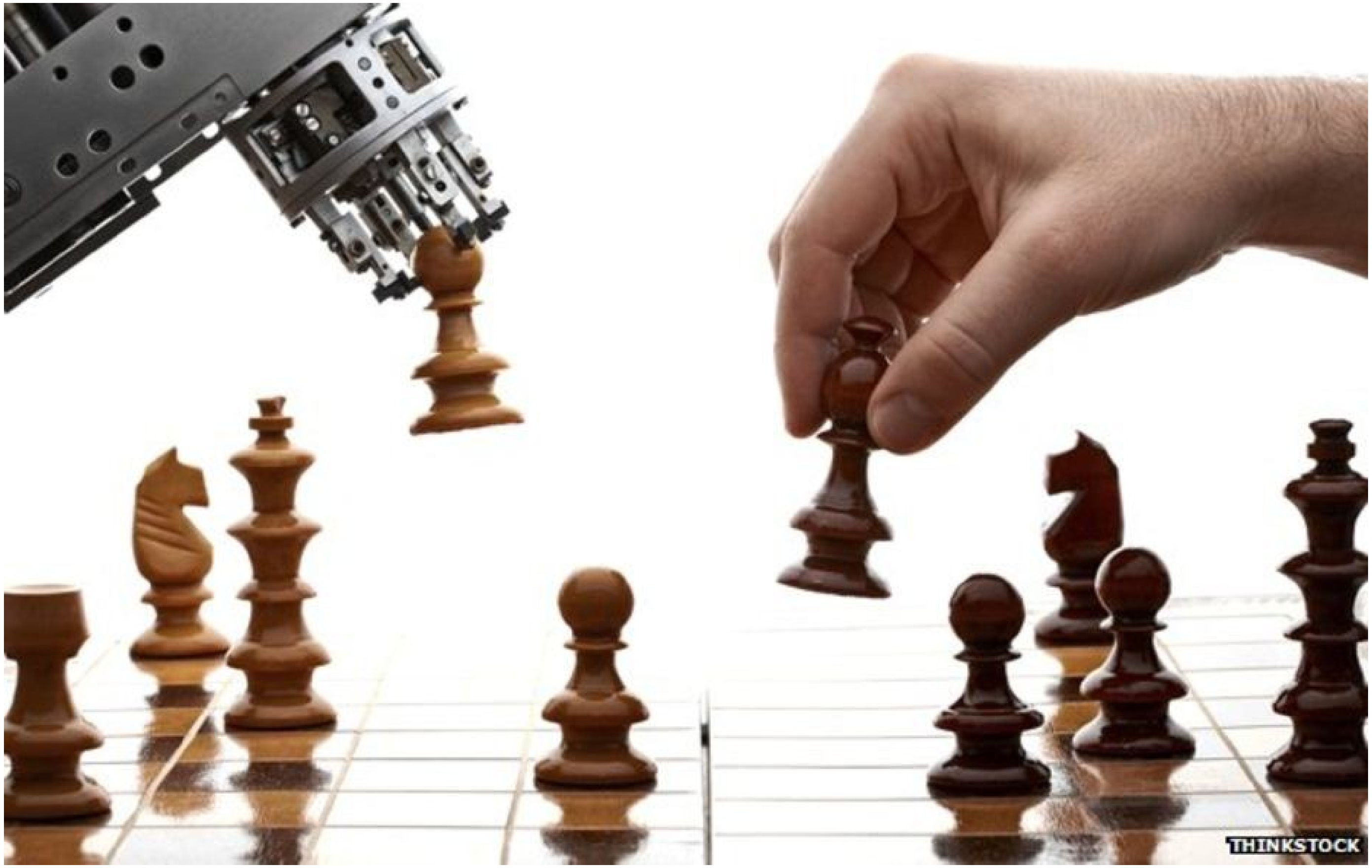


Introduction to Artificial Intelligence

COSC 4550 / COSC 5550

Professor Cheney

10/4/17

hidden Markov models

a Markov process where the state
is a single discrete variable

our example from last class was one of these
(the world was described only by whether it rained or not)

having a single state variable allows for convenient representation of the transition model as a matrix (T)

$$T = P(X_t \mid X_{t-1}) = P(Rain_t \mid Rain_{t-1}) = \begin{array}{c|cc} & r_t = t & r_t = f \\ \hline r_{t-1} = t & 0.7 & 0.3 \\ r_{t-1} = f & 0.3 & 0.7 \end{array}$$

you can similarly represent the change of observing your evidence variables as a sensor model matrix (O)

if $u_t = t$:

$$O = P(E_t \mid X_t) = P(Umbrella_t \mid Rain_t) = \begin{matrix} r_t = t \\ r_t = f \end{matrix} \begin{bmatrix} 0.9 & 0 \\ 0 & 0.2 \end{bmatrix}$$

if $u_t = f$:

$$\begin{matrix} r_t = t \\ r_t = f \end{matrix} \begin{bmatrix} 0.1 & 0 \\ 0 & 0.8 \end{bmatrix}$$

(diagonal matrix with $P(e_t \mid X_t = i)$ on main diagonal) this matrix representation let's us compactly represent and compute our forward and backward passes

$$f_{1:t+1} = \alpha\, O_{t+1}\, T^T f_{1:t}$$

$$b_{k+1:t} = T\, O_{k+1}\, b_{k+2:t}$$

$P(R_1 \mid u_1)$ → $f_{1:t+1}$; $P(R_0)$ → $f_{1:t}$

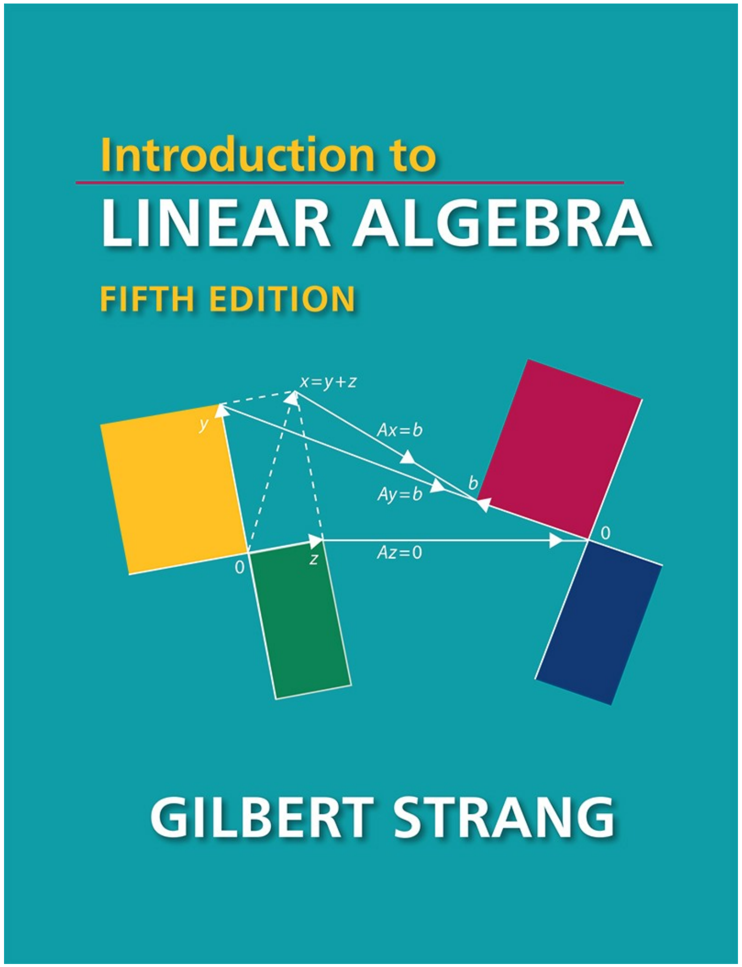
$$f_{1:t+1} = \alpha\, O_{t+1}\, T^T f_{1:t} = \alpha \begin{bmatrix} 0.9 & 0 \\ 0 & 0.2 \end{bmatrix} \begin{bmatrix} 0.7 & 0.3 \\ 0.3 & 0.7 \end{bmatrix} \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix} = \begin{bmatrix} 0.818 \\ 0.182 \end{bmatrix}$$

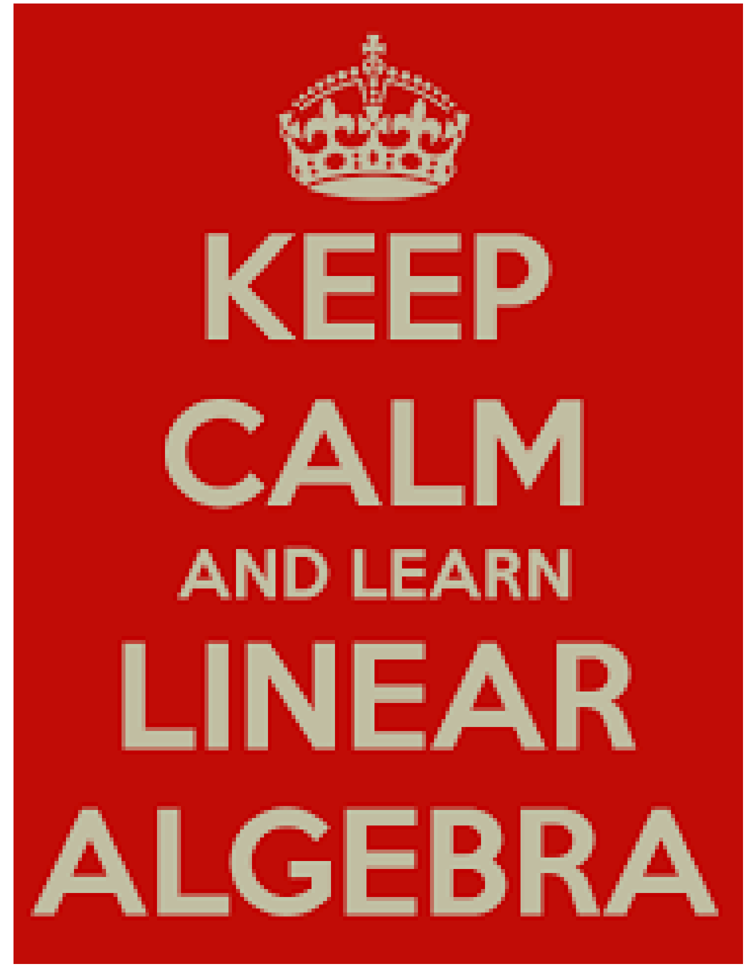
lots of other convenient (time and space saving) tricks you can do with matrix representations (some in your book!)

like the first-order Markov process trick,
you can get around the single variable constraint
by creating a "megavariable" (tuple of any length)
and having each combination of states be a megastate

T =

	$(s_t = t, r_t = t)$	$(s_t = t, r_t = f)$	$(s_t = f, r_t = t)$	$(s_t = f, r_t = f)$
$(summer_{t-1} = t, r_{t-1} = t)$				
$(summer_{t-1} = t, r_{t-1} = f)$				
$(summer_{t-1} = f, r_{t-1} = t)$				
$(summer_{t-1} = f, r_{t-1} = f)$				


Introduction to
LINEAR ALGEBRA
FIFTH EDITION
x=y+z
y
Ax=b
Ay=b
b
0
z
Az=0
0
GILBERT STRANG


KEEP
CALM
AND LEARN
LINEAR
ALGEBRA

Kalman filters

so far we've been iterating over the possible values of discrete variables over time with HMMs

but many interesting real world problems are continuous

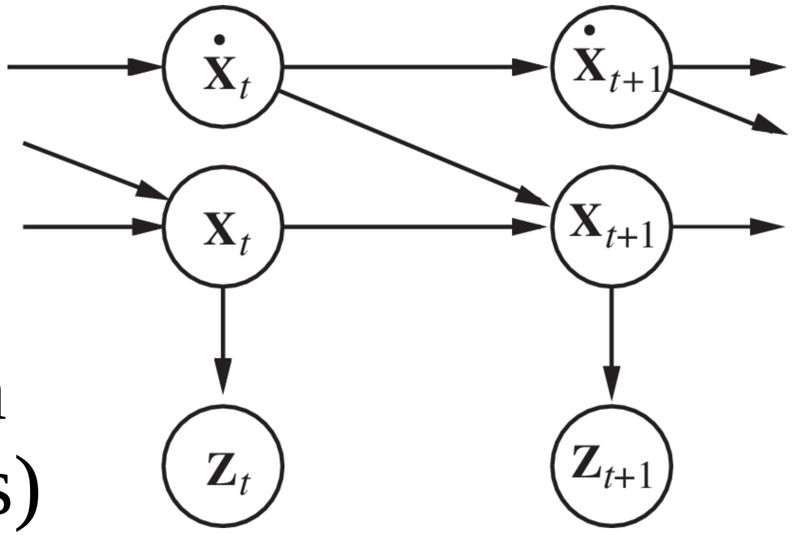
and partially observable and/or noisy measurements (and models)

Kalman filters can deal with these types of problems (so they are used very widely in practice!)

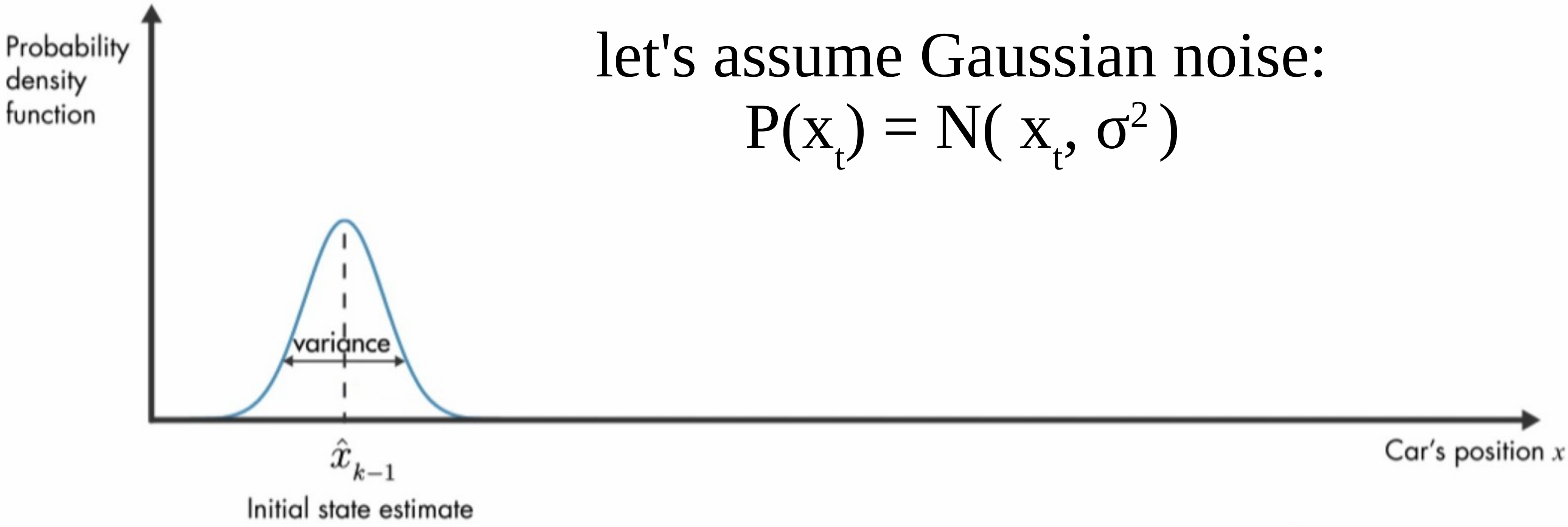
let's say we are interested in tracking the movement over time of something (car, plane, bird, ball, missle, …)

our model to track the movement might incorporate information about the object's 3D position and velocity our (linear) transition model:

$$X_{t+\Delta} = X_t + X' * \Delta$$



let's assume imperfect information
(noisy measurements/state estimates)



let's assume Gaussian noise:

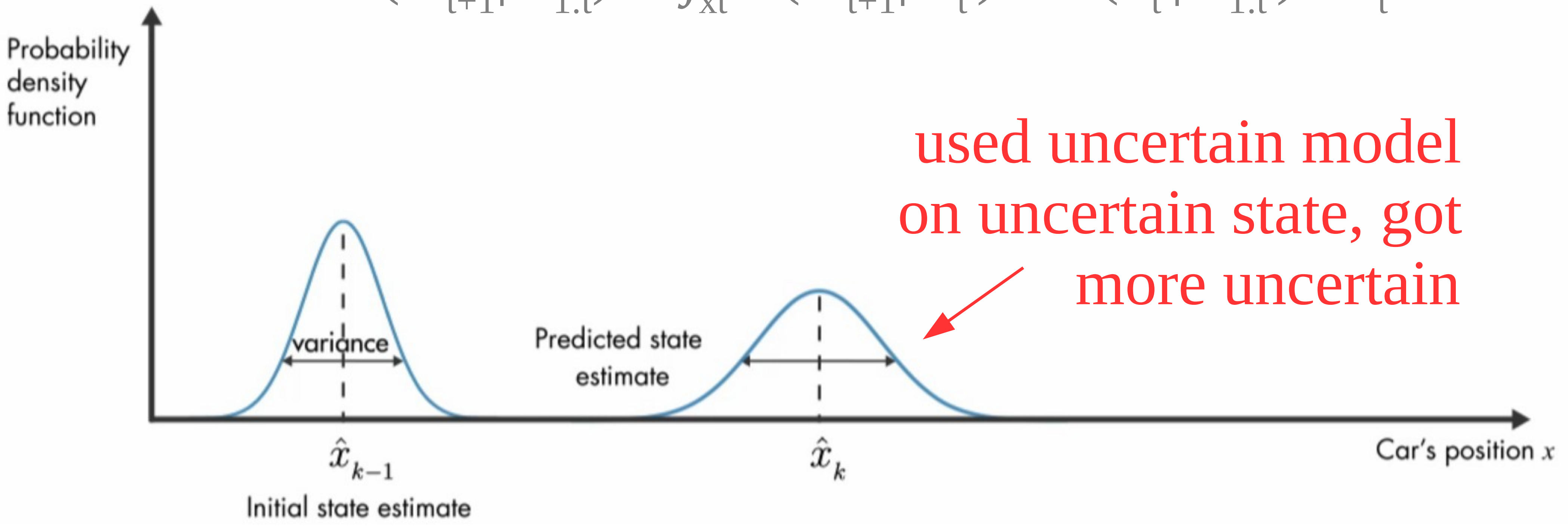
$$P(x_t) = N(x_t, \sigma^2)$$

let's also assume a noisy model (also w/ Gaussian noise):

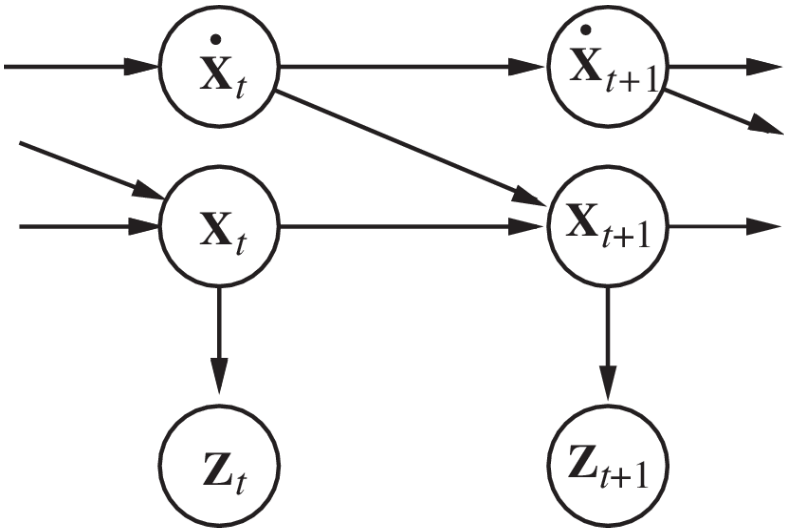
$$P(x_{t+\Delta} | x_t, x'_t) = N(x_t + x' * \Delta, \sigma^2)(x_{t+\Delta})$$

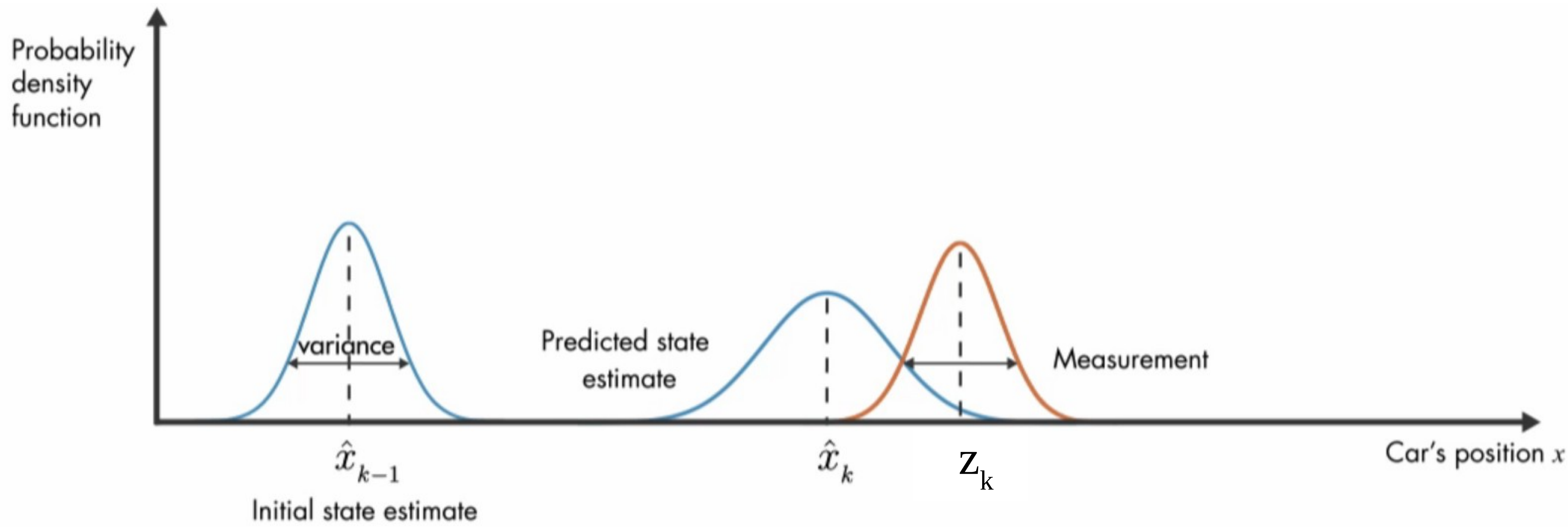
side note, since we these are continuous variables, all probability functions are integrals:

$$P(X_{t+1} | e_{1:t}) = \int_{xt} P(X_{t+1} | x_t) * P(x_t | e_{1:t})\, dx_t$$

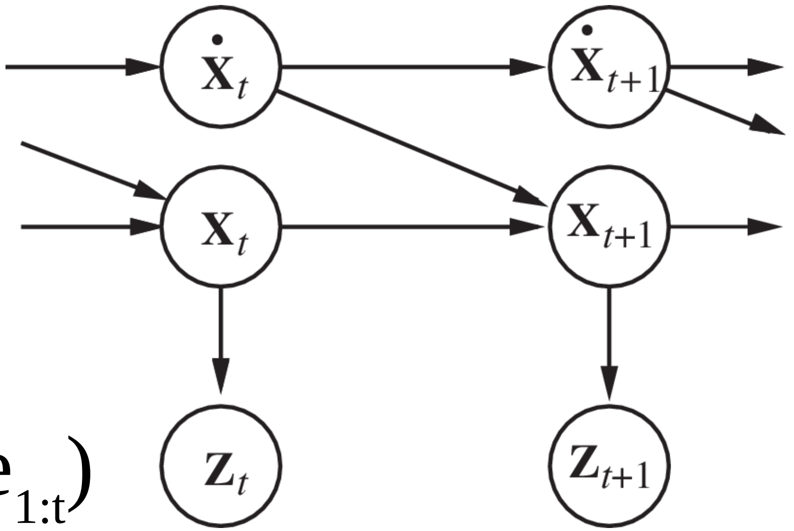


our observation at the next timestep provides new information about the world, but is also uncertain (and biased)

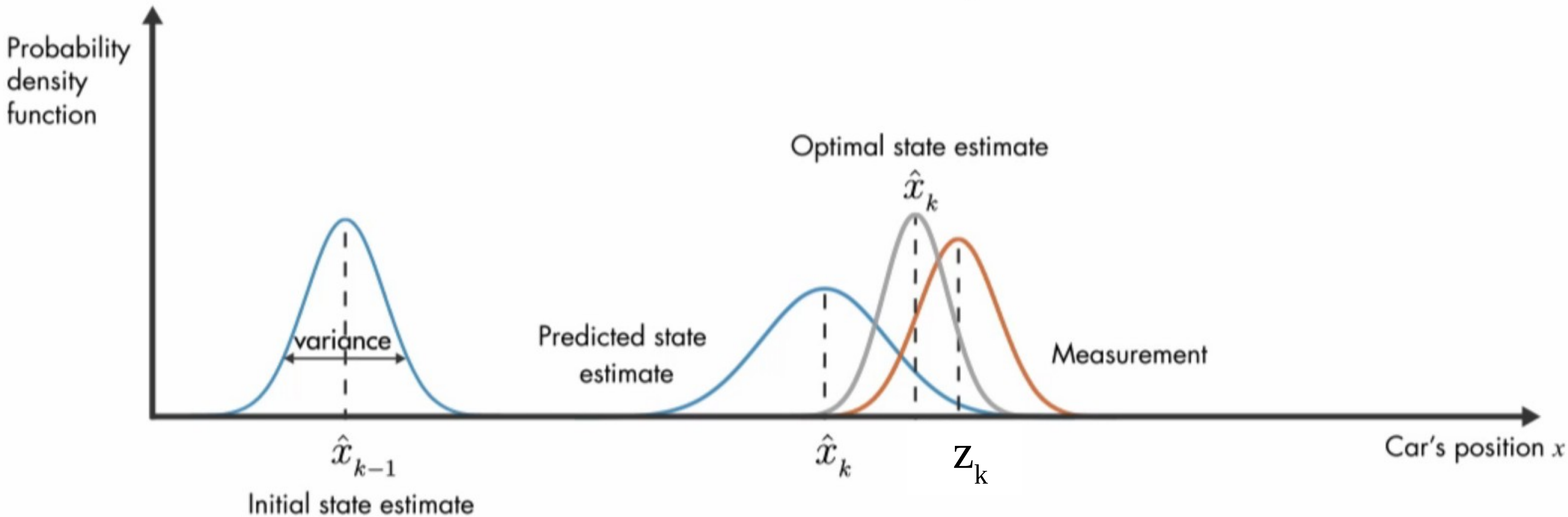




combining the forward prediction
with our new observation
give us a more informed estimate



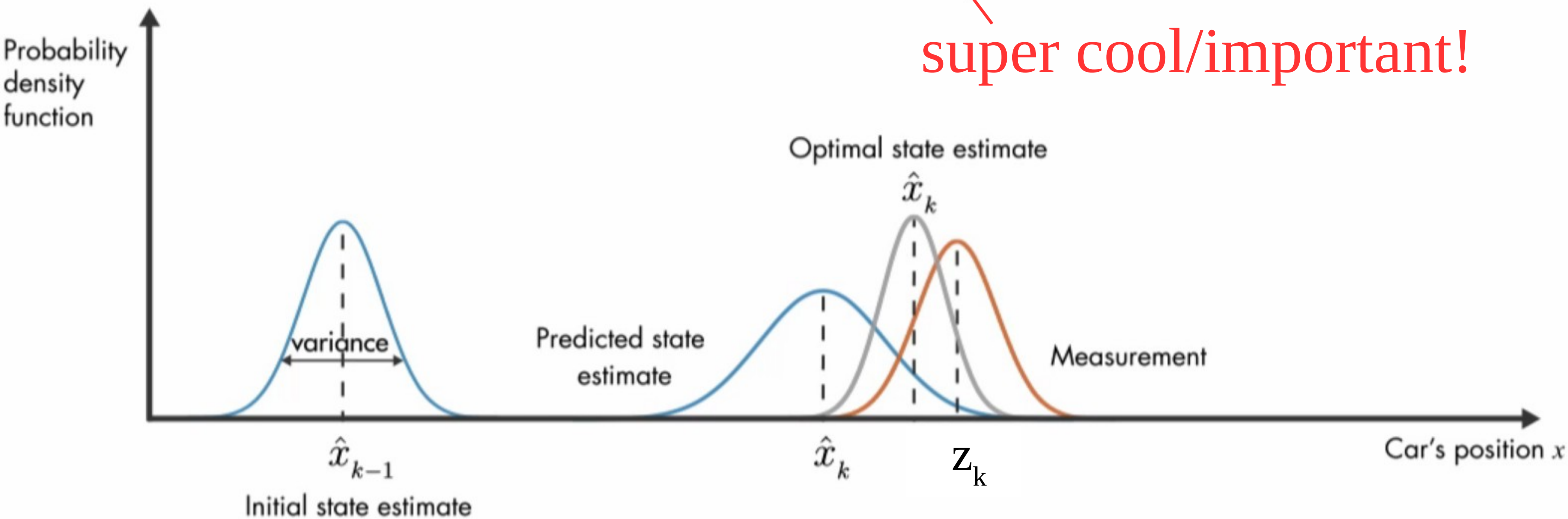
$$P(X_{t+1} | e_{1:t+1}) = \alpha\, P(e_{t+1} | X_{t+1}) * P(X_{t+1} | e_{1:t})$$

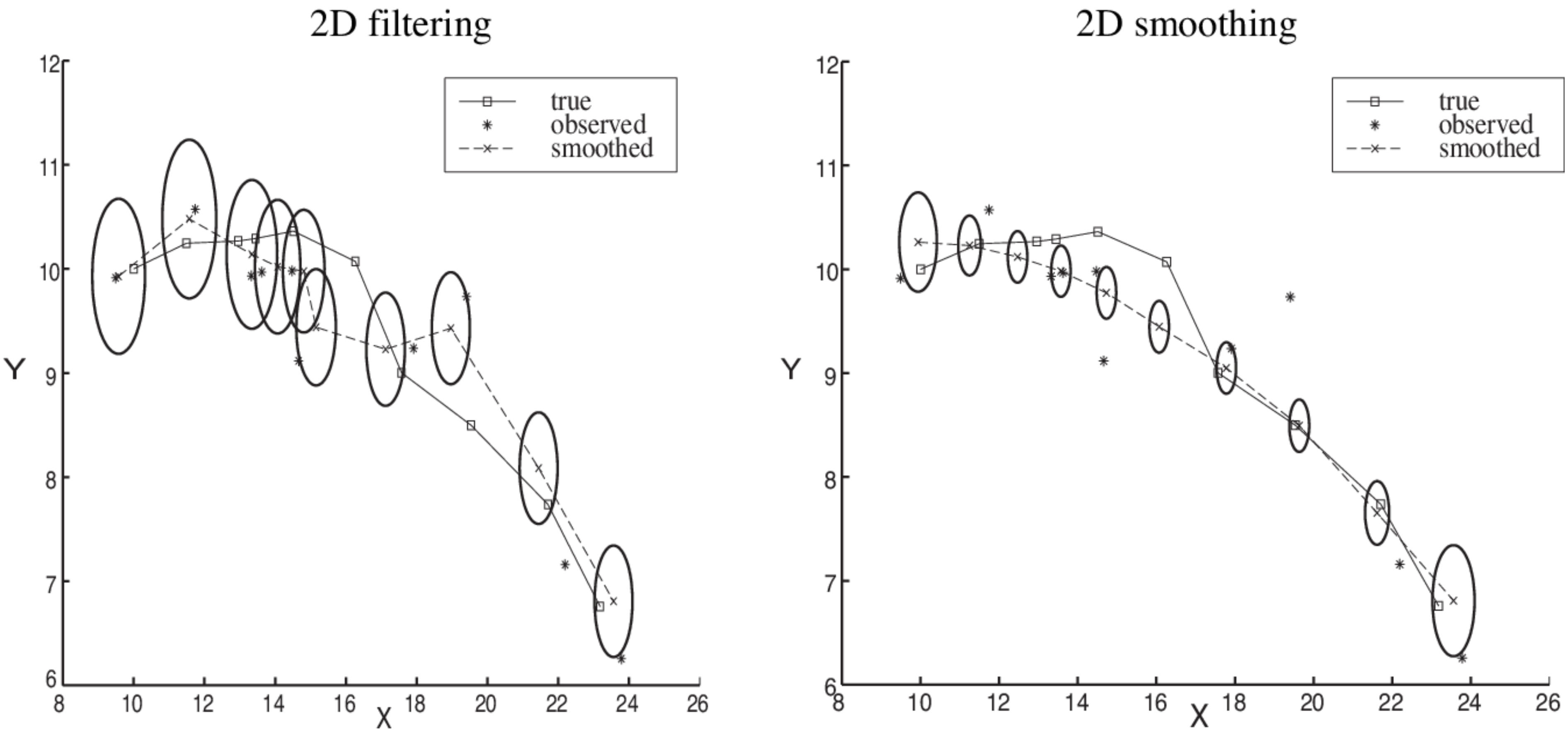


the variance of our new estimate is less than either our predicted state or our measurement (combining two estimates gives us more certainty!)

this is a property of multiplying Gaussian distribution (variance often grows when multiplying other distributions)

super cool/important!




2D filtering
true
observed
smoothed
Y
X
12
11
10
9
8
7
6
8
10
12
14
16
18
20
22
24
26
2D smoothing
true
observed
smoothed
Y
X
12
11
10
9
8
7
6
8
10
12
14
16
18
20
22
24
26