EDUARDO VELLOSO

Mid-Semester

REVIEW

What we've covered so far:

- Research methods
- Mobile interaction
- Sensor data processing
- Activity recognition

ACTIVITY PREP

Make groups of 3 students
Assign a letter to each student

ACTIVITY 1

Individually, draft a UX research plan for deriving a set of activities to be recognised by the system using the corresponding method

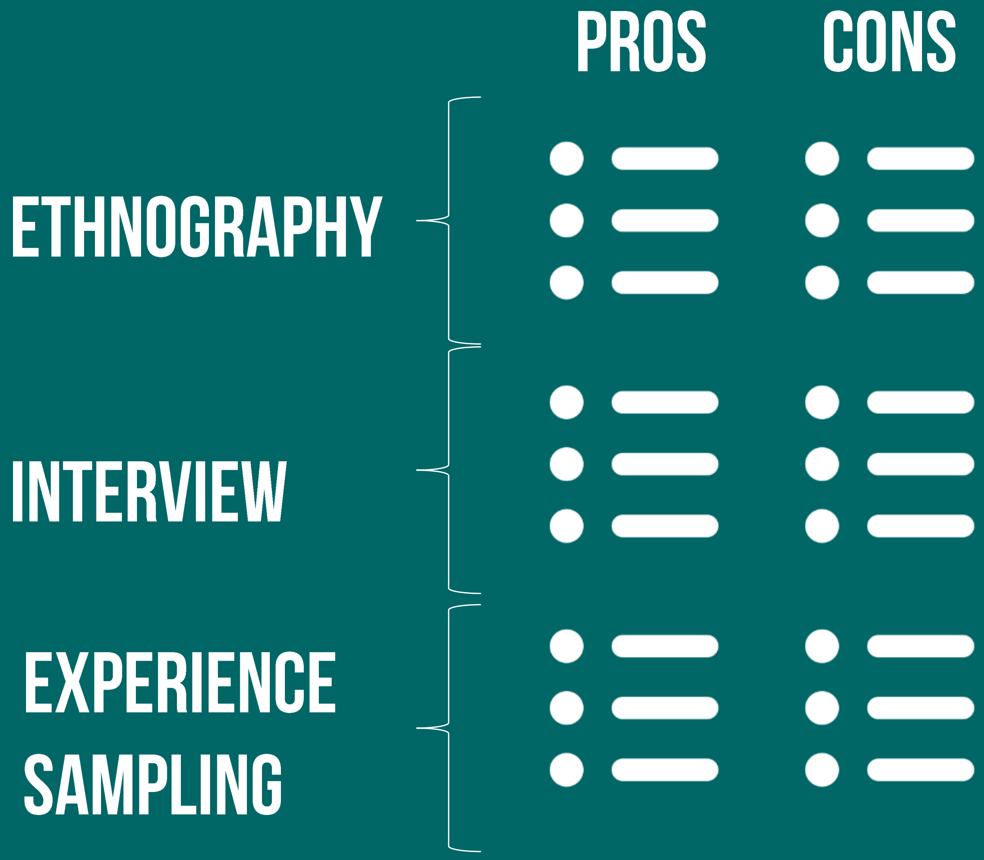
A ETHNOGRAPHY

B INTERVIEW

C EXPERIENCE SAMPLING

ACTIVITY 2

Present your ideas to your colleagues
As a group, make a Pros and Cons list for each method.
Reach a consensus on which one to use.



RESULTS:

(Open, Close) Bonnet
(Open, Close, Check) Boot
(Open, Close) (Left, Right) Door
(Open, Close) Fuel Lid
(Open, Close) Spare wheel box
Writing
Check mirror

ACTIVITY 3

Create a confusion matrix ordering the rows and columns by class similarity

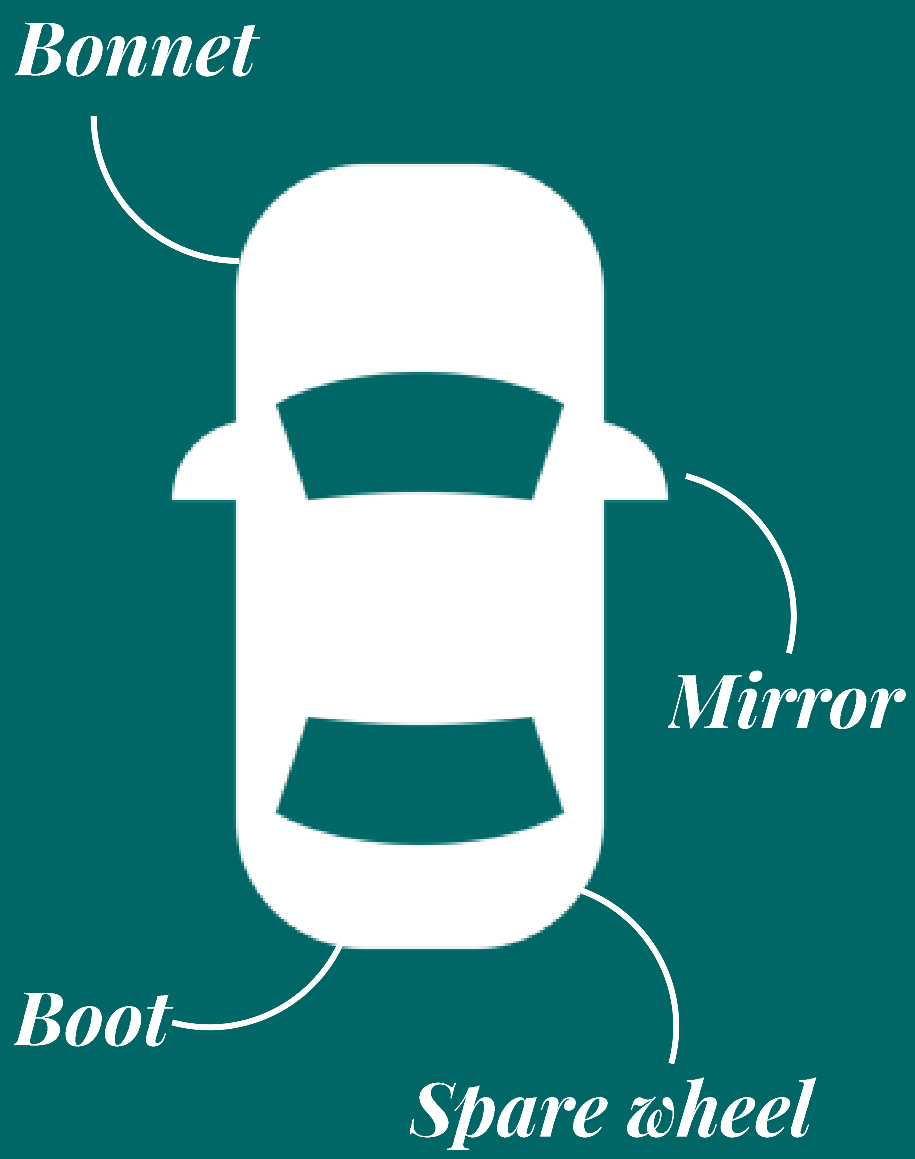
Given 100 samples in each class, estimate the number in each cell

Output

Ground truth

CLASSES:

(Open, Close) Bonnet
(Open, Close, Check) Boot
(Open, Close) (Left, Right) Door
(Open, Close) Fuel Lid
(Open, Close) Spare wheel box
Writing
Check mirror



ACTIVITY 4

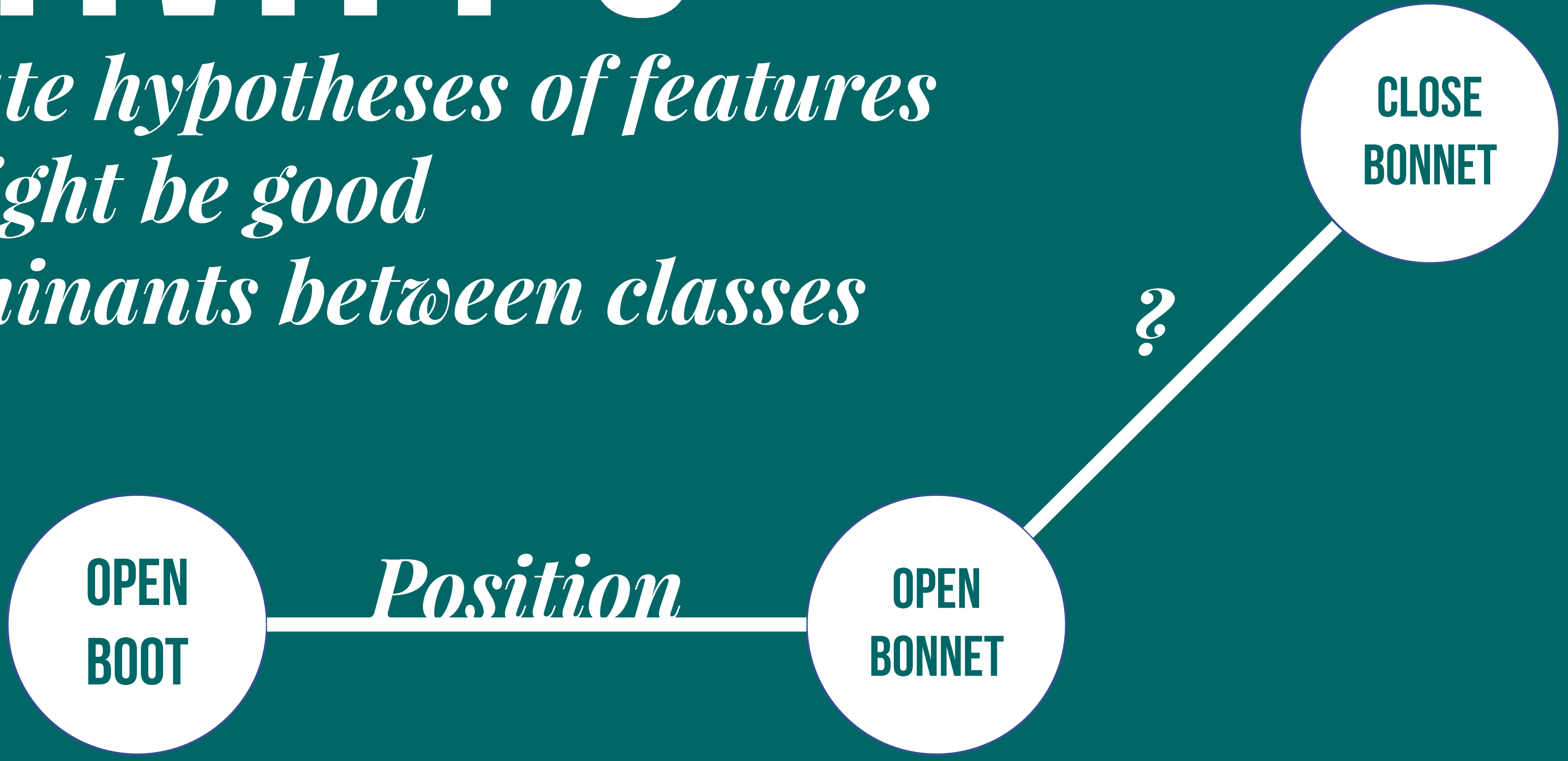
Based on your estimates, calculate the precision, recall, and F1 score for the classes 'close boot' and 'writing'

Output

Ground truth

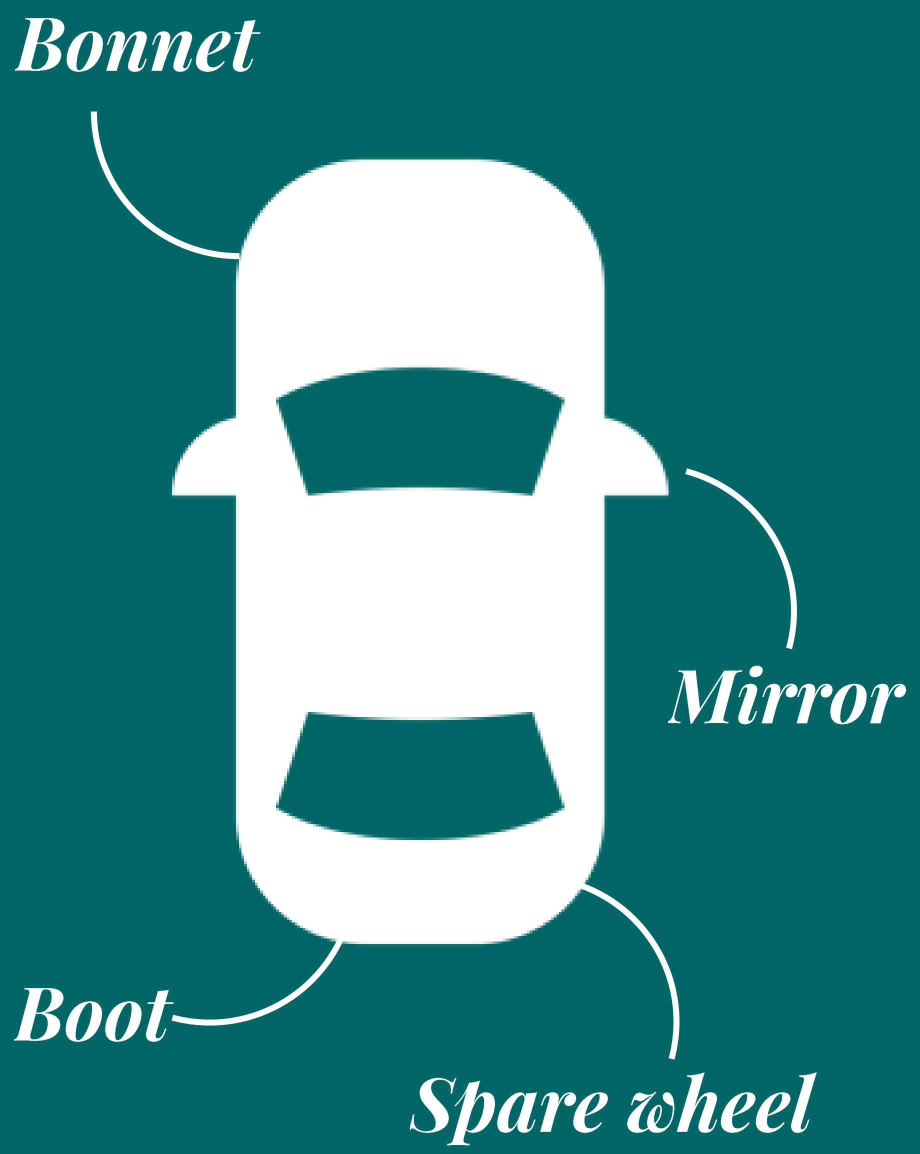
ACTIVITY 5

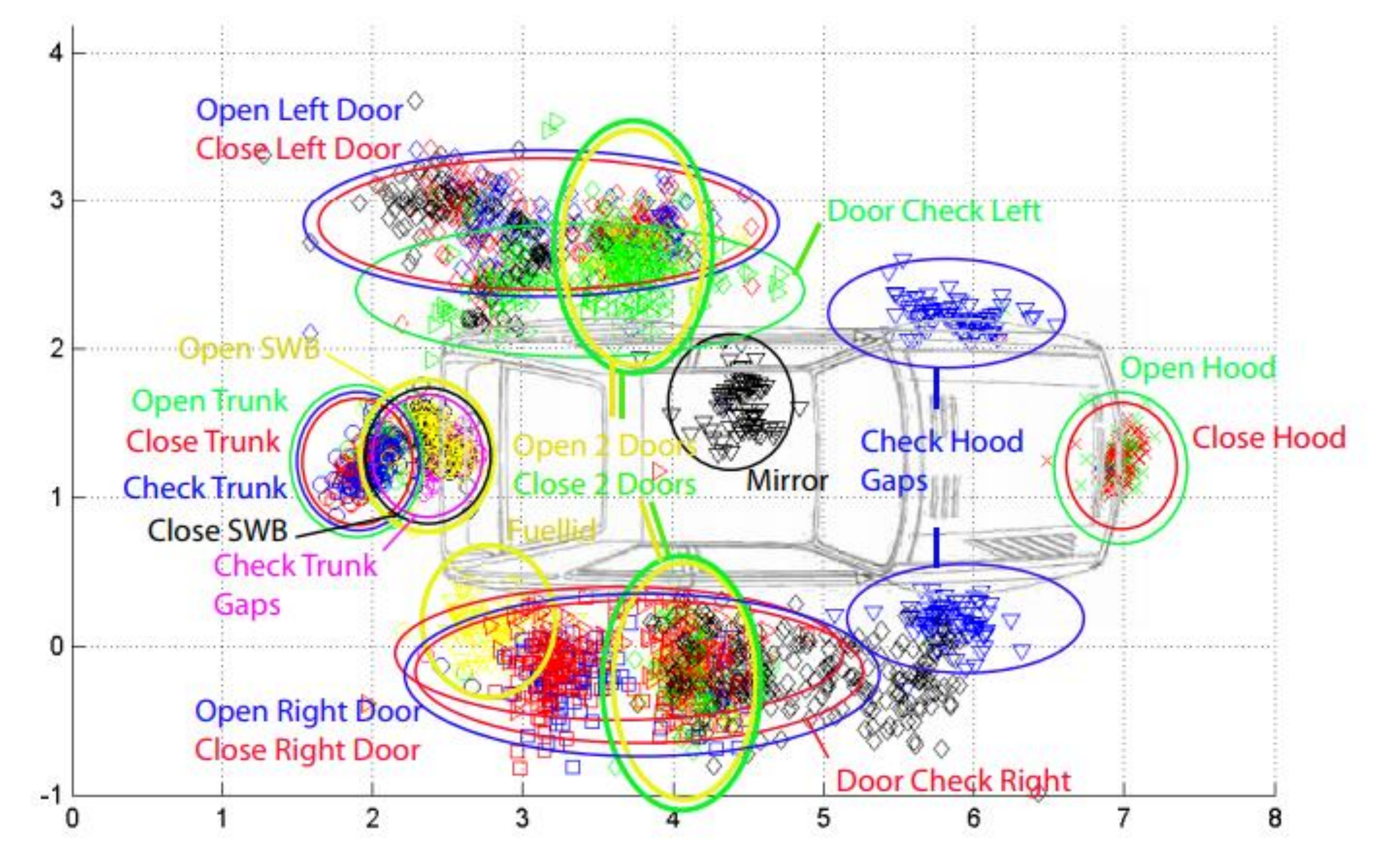
Generate hypotheses of features that might be good discriminants between classes



CLASSES:

(Open, Close) Bonnet
(Open, Close, Check) Boot
(Open, Close) (Left, Right) Door
(Open, Close) Fuel Lid
(Open, Close) Spare wheel box
Writing
Check mirror




Open Left Door
Close Left Door
Door Check Left
Open SWB
Open Trunk
Close Trunk
Check Trunk
Close SWB
Check Trunk
Gaps
Open 2 Doors
Close 2 Doors
Fuellid
Mirror
Check Hood
Gaps
Open Hood
Close Hood
Open Right Door
Close Right Door
Door Check Right
4
3
2
1
0
-1
0
1
2
3
4
5
6
7
8

ACTIVITY 6

Choose sensors and devices that could provide the data for the features that you chose.

e.g. position ➡ beacons + phone

3.3 Location Features

The worker's relative position to the car body is estimated using an ultra-wide band (UWB) system by Ubisense. Sensor data of four tags is smoothed with a Kalman filter before the four sensor sources are combined. Despite the use of multiple sensors, the data becomes patchy at certain times. In a preprocessing step, we estimate location values for these gaps by linearly interpolating between known values in time. Based on these location values, two different location rep-

measurement

$z_i = ?$

state

$x_i = ?$

Robot only moves in a line

We want to keep track of its:

- *Position*
- *Velocity*

The robot has a position and a velocity sensors

measurement

$$z_i = \begin{pmatrix} z_i^{(x)} \\ z_i^{(v)} \end{pmatrix}$$

Robot only moves in a line
We want to keep track of its:
- Position
- Velocity
The robot has a position and a velocity sensors

state

$$x_i = \begin{pmatrix} x_i \\ v_i \end{pmatrix}$$

measurement

$$z_i = \begin{pmatrix} z_i^{(x)} \\ z_i^{(y)} \end{pmatrix}$$

state

$$x_i = \begin{pmatrix} x_i \\ y_i \\ v_i^{(x)} \\ v_i^{(y)} \end{pmatrix}$$

From state to measurement

$$\boldsymbol{z_i} = \boldsymbol{H}\,\boldsymbol{x_i} + \boldsymbol{v_i}$$

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

$$v_i \sim N(0, \boldsymbol{R_i})$$

$$R_i = \begin{bmatrix} \sigma^2 & 0 \\ 0 & \sigma^2 \end{bmatrix}$$

measurement

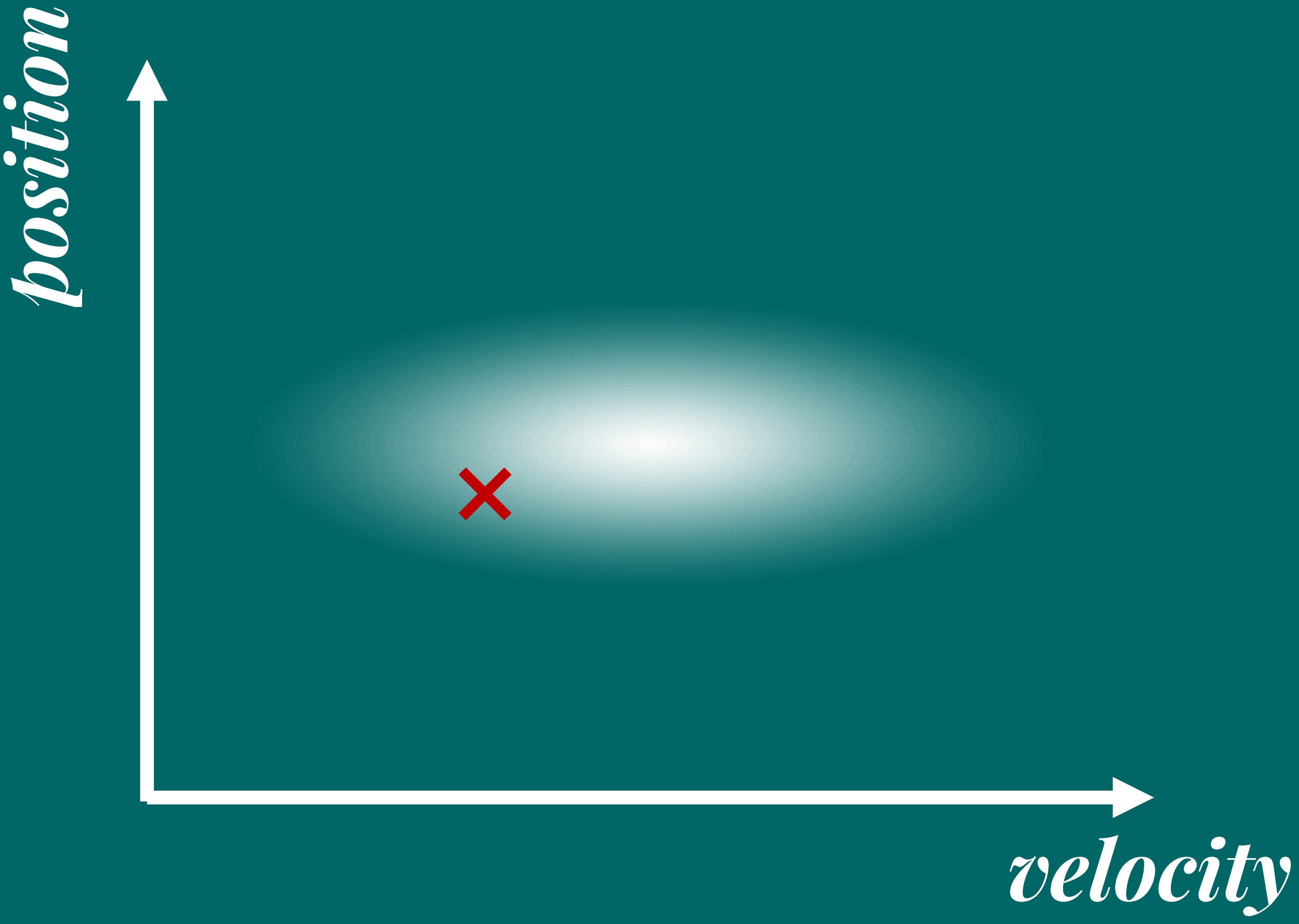
$$z_i = \begin{pmatrix} z_i^{(x)} \\ z_i^{(v)} \end{pmatrix}$$

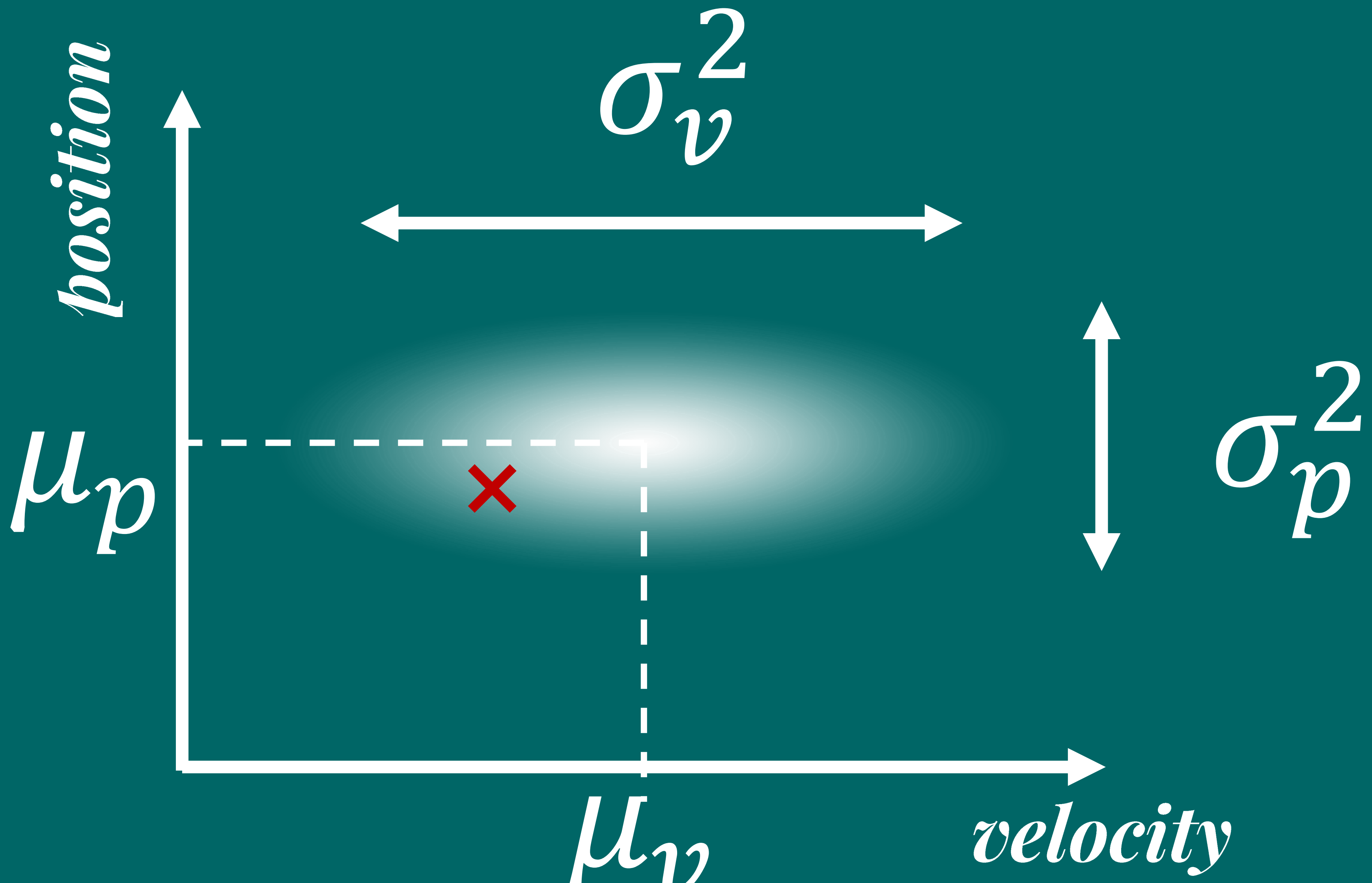
state

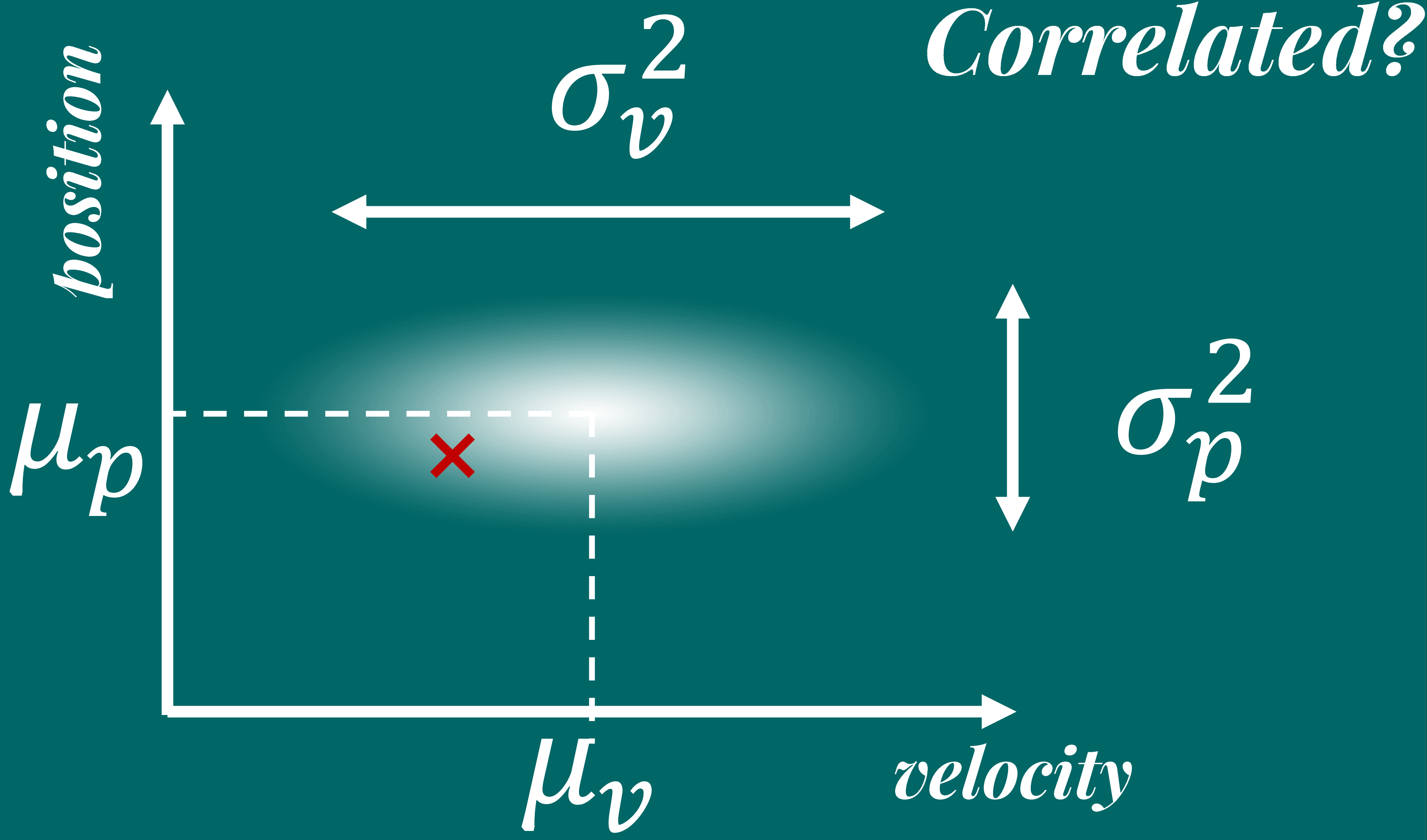
$$x_i = \begin{pmatrix} x_i \\ v_i \end{pmatrix}$$

$$\boldsymbol{z_i} = \boldsymbol{H}\,\boldsymbol{x_i} + \boldsymbol{v_i}$$

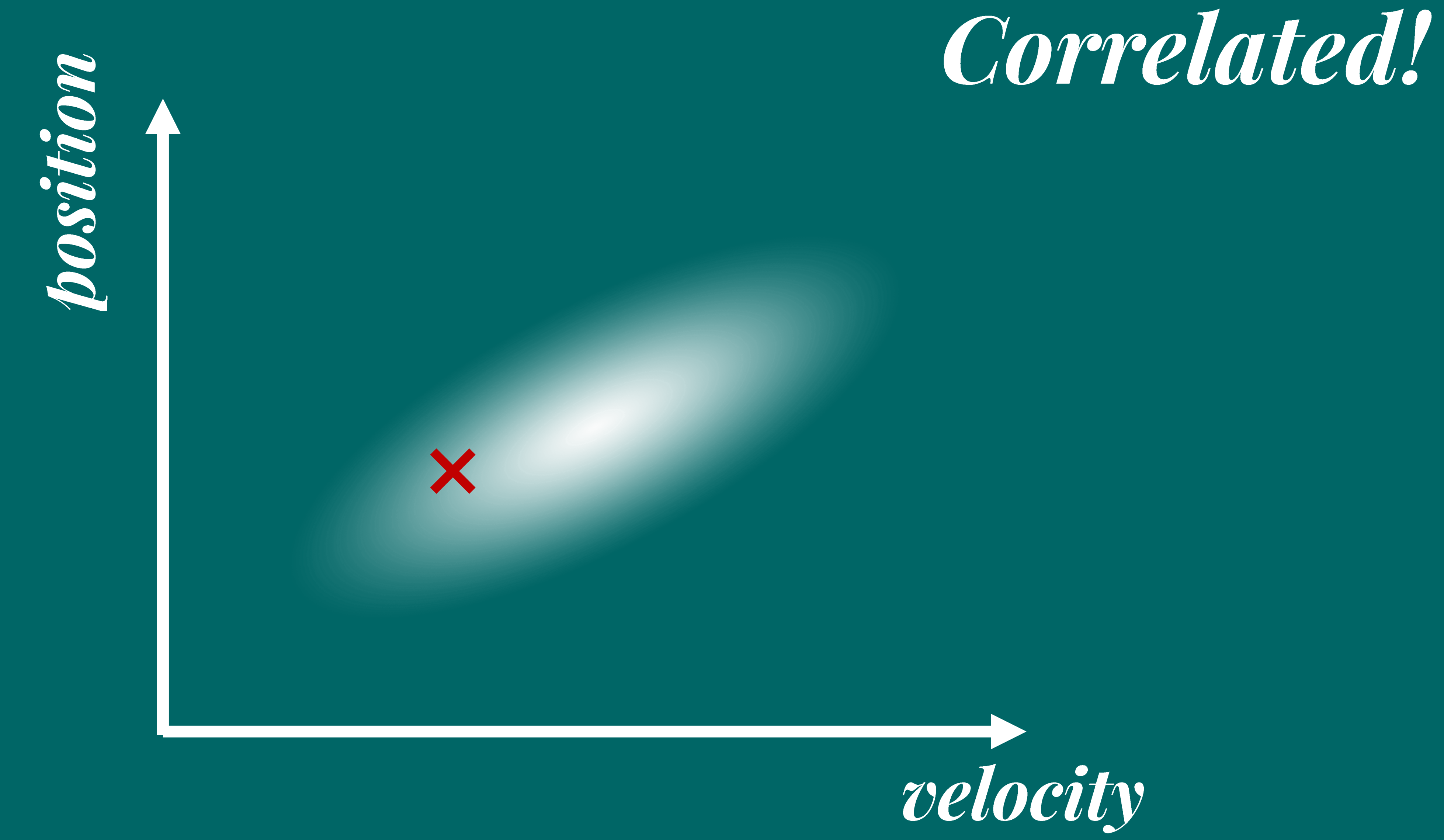
$$H = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$


position
velocity


σ_v^2
position
μ_p
σ_p^2
μ_v
velocity



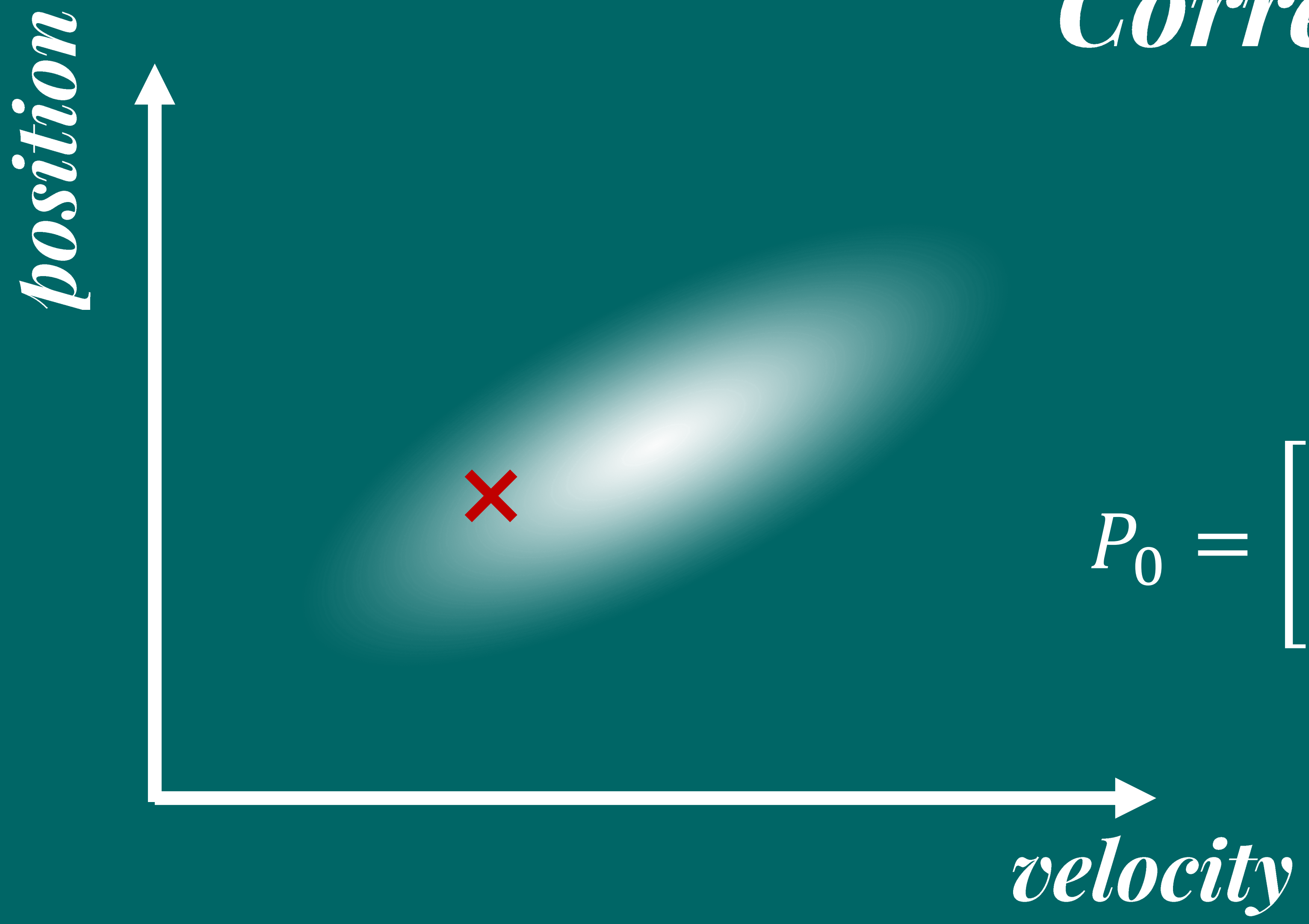
Correlated?
σ_v^2
position
σ_p^2
μ_p
μ_v
velocity


Correlated!
position
velocity

In the GPS Example....

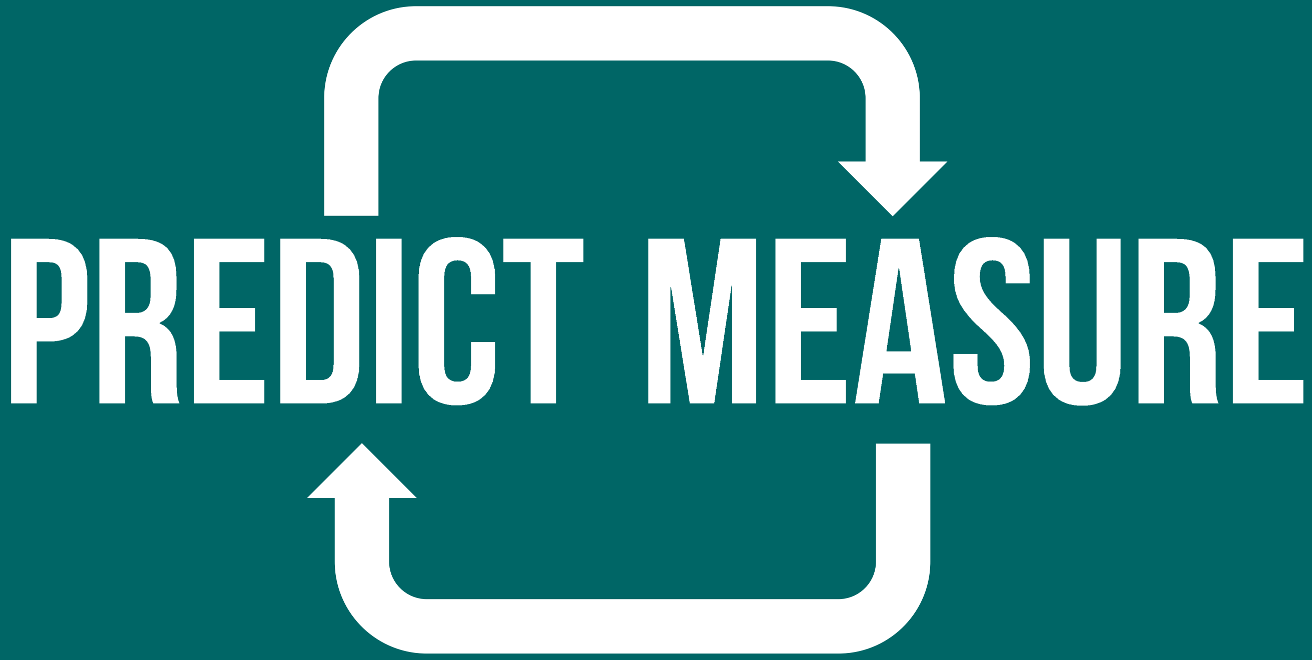
$$P_0 = \begin{bmatrix} \sigma^2 & 0 & 0 & 0 \\ 0 & \sigma^2 & 0 & 0 \\ 0 & 0 & \sigma_s^2 & 0 \\ 0 & 0 & 0 & \sigma_s^2 \end{bmatrix}$$

Correlated!

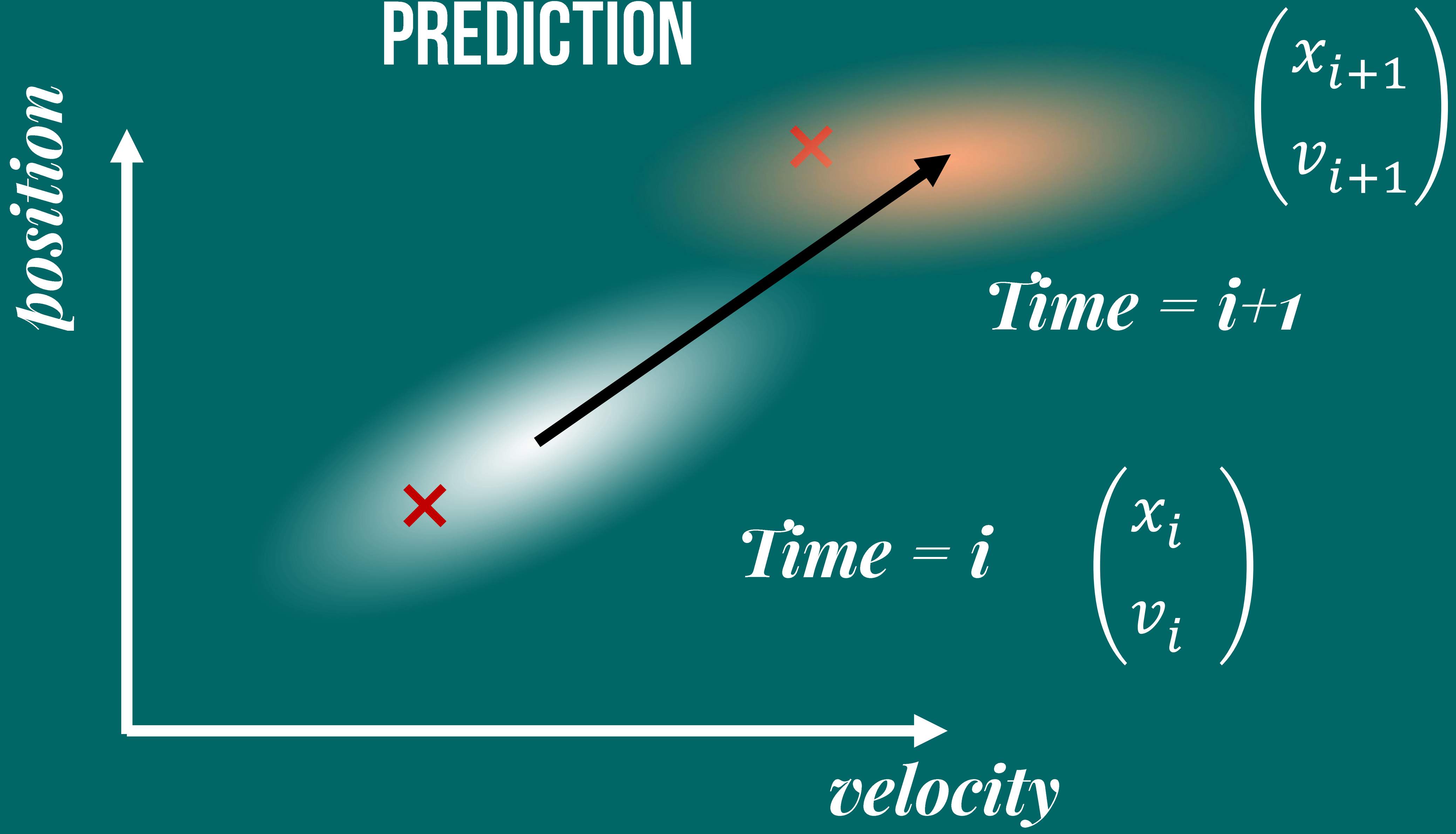


$$P_0 = \begin{bmatrix} \sigma_p^2 & \sigma_{pv}^2 \\ \sigma_{pv}^2 & \sigma_v^2 \end{bmatrix}$$

THE ALGORITHM



PREDICTION


position
velocity
Time = i
$\begin{pmatrix} x_i \\ v_i \end{pmatrix}$
Time = i+1
$\begin{pmatrix} x_{i+1} \\ v_{i+1} \end{pmatrix}$

$$\begin{pmatrix} x_{i+1} \\ v_{i+1} \end{pmatrix} = \underbrace{\begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix}}_{\phi} \begin{pmatrix} x_i \\ v_i \end{pmatrix}$$

From statistics:

Say x is a distribution. P is its covariance.

$$Cov(x) = P$$

But if we multiply every value in x by A…

$$Cov(Ax) = APA^T$$

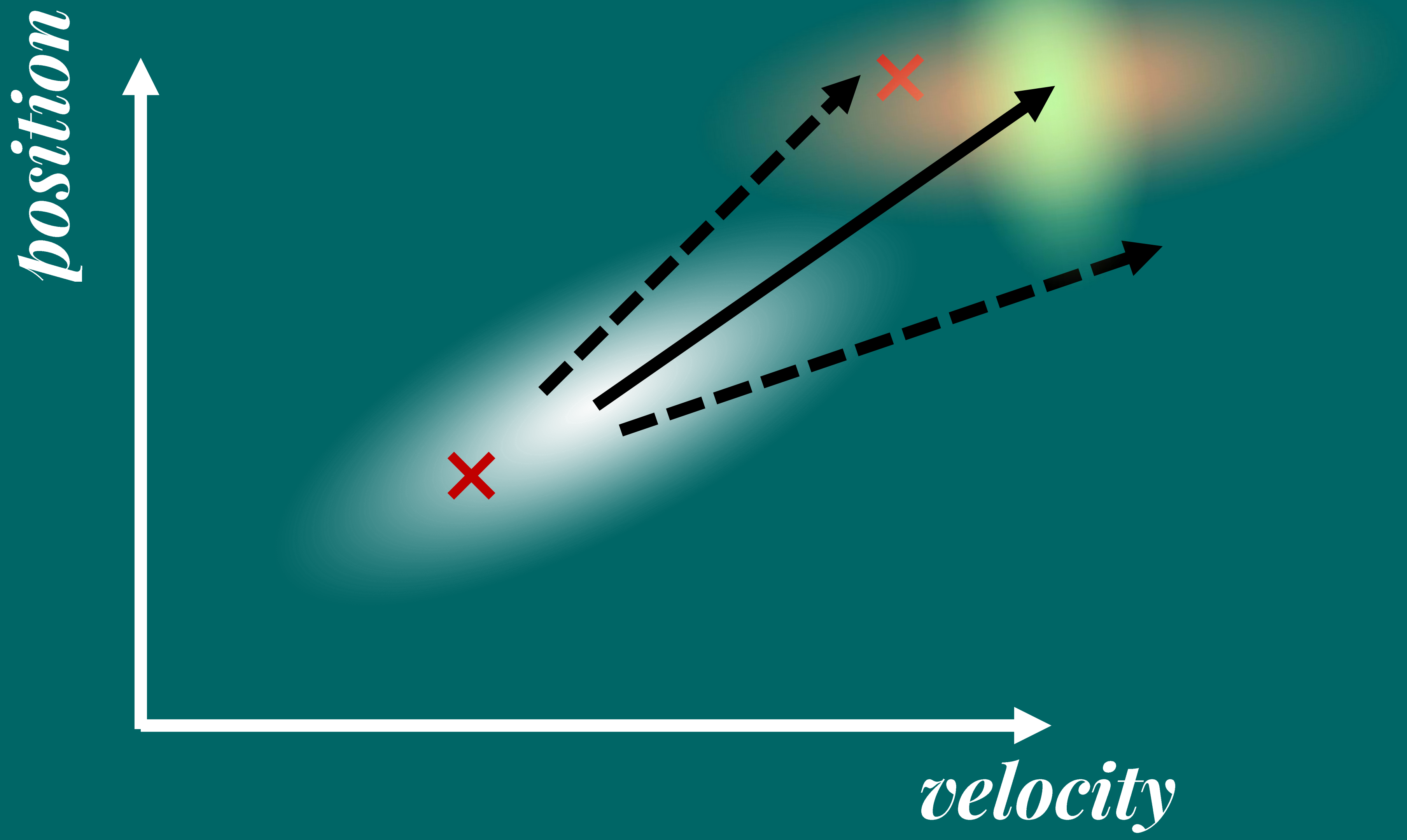
$$\begin{pmatrix} x_{i+1} \\ v_{i+1} \end{pmatrix} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \begin{pmatrix} x_i \\ v_i \end{pmatrix}$$

$$x_{i+1} = \phi x_i$$

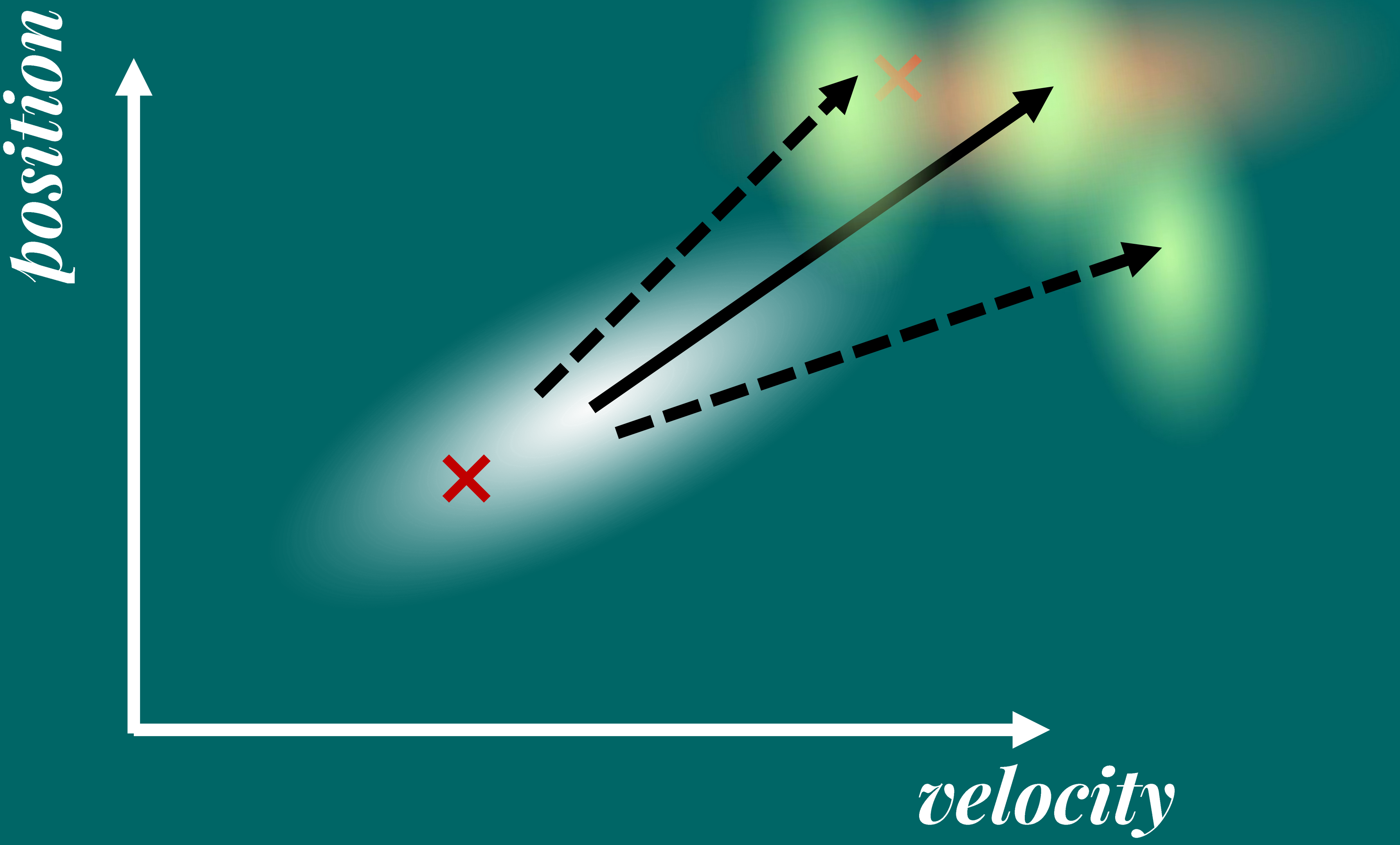
$$P_{i+1} = \phi P_i \phi^T$$

Because P is the covariance of x!

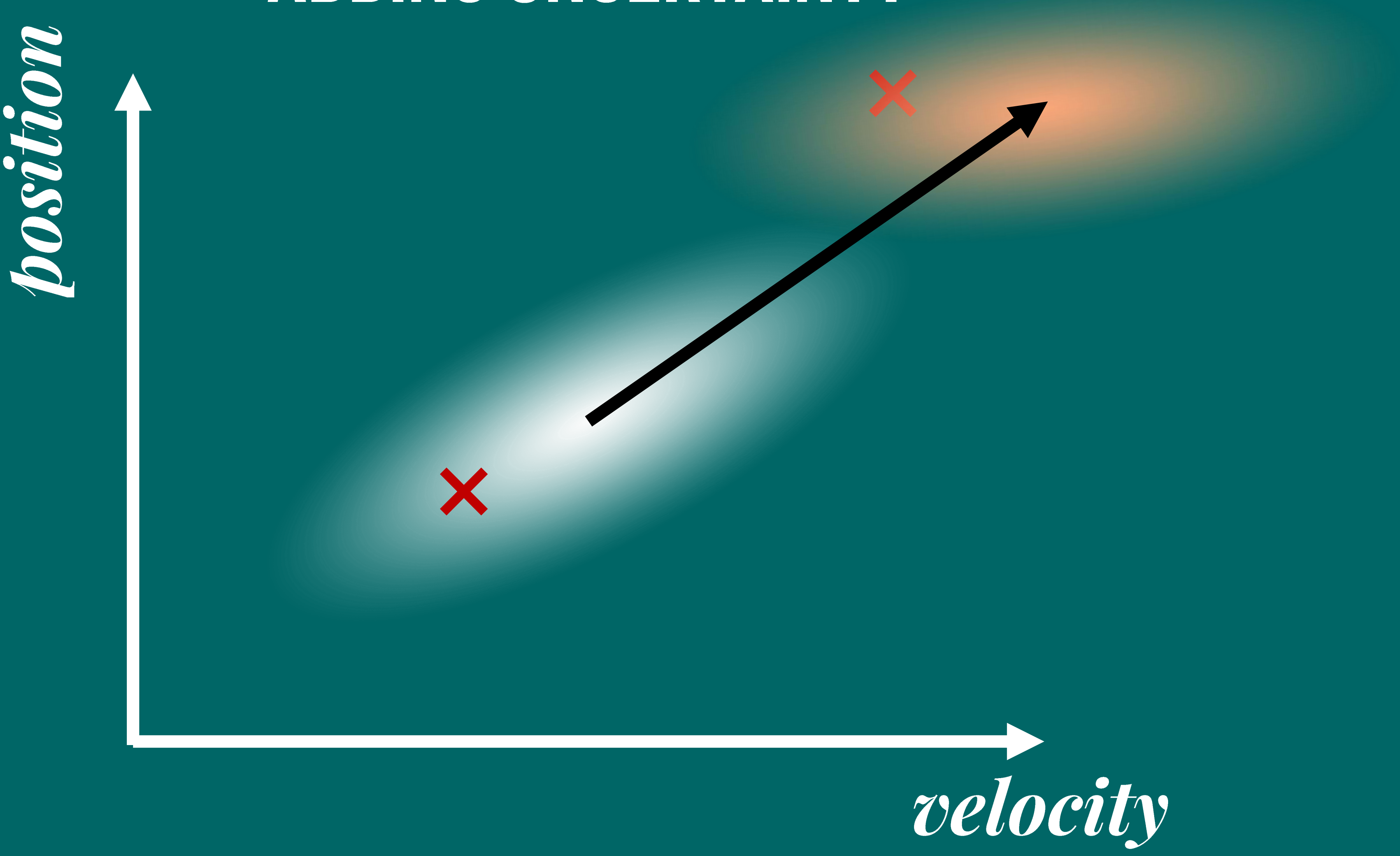
ADDING UNCERTAINTY



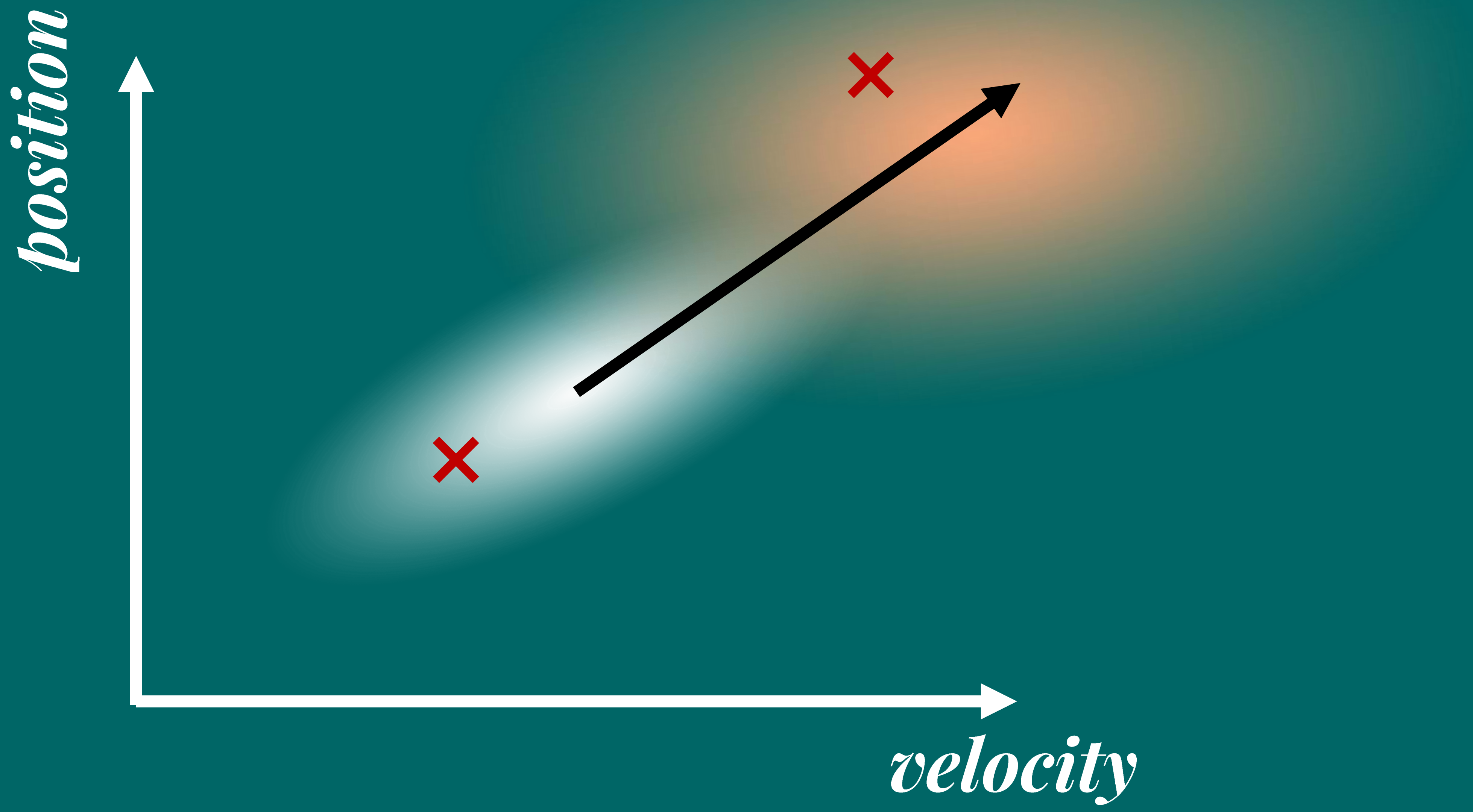
ADDING UNCERTAINTY



ADDING UNCERTAINTY



ADDING UNCERTAINTY



$$\begin{pmatrix} x_{i+1} \\ v_{i+1} \end{pmatrix} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \begin{pmatrix} x_i \\ v_i \end{pmatrix}$$

$$x_{i+1} = \phi x_i$$

$$P_{i+1} = \phi P_i \phi^T + Qi$$

Because P is the covariance of x!

STEP 1: PREDICT

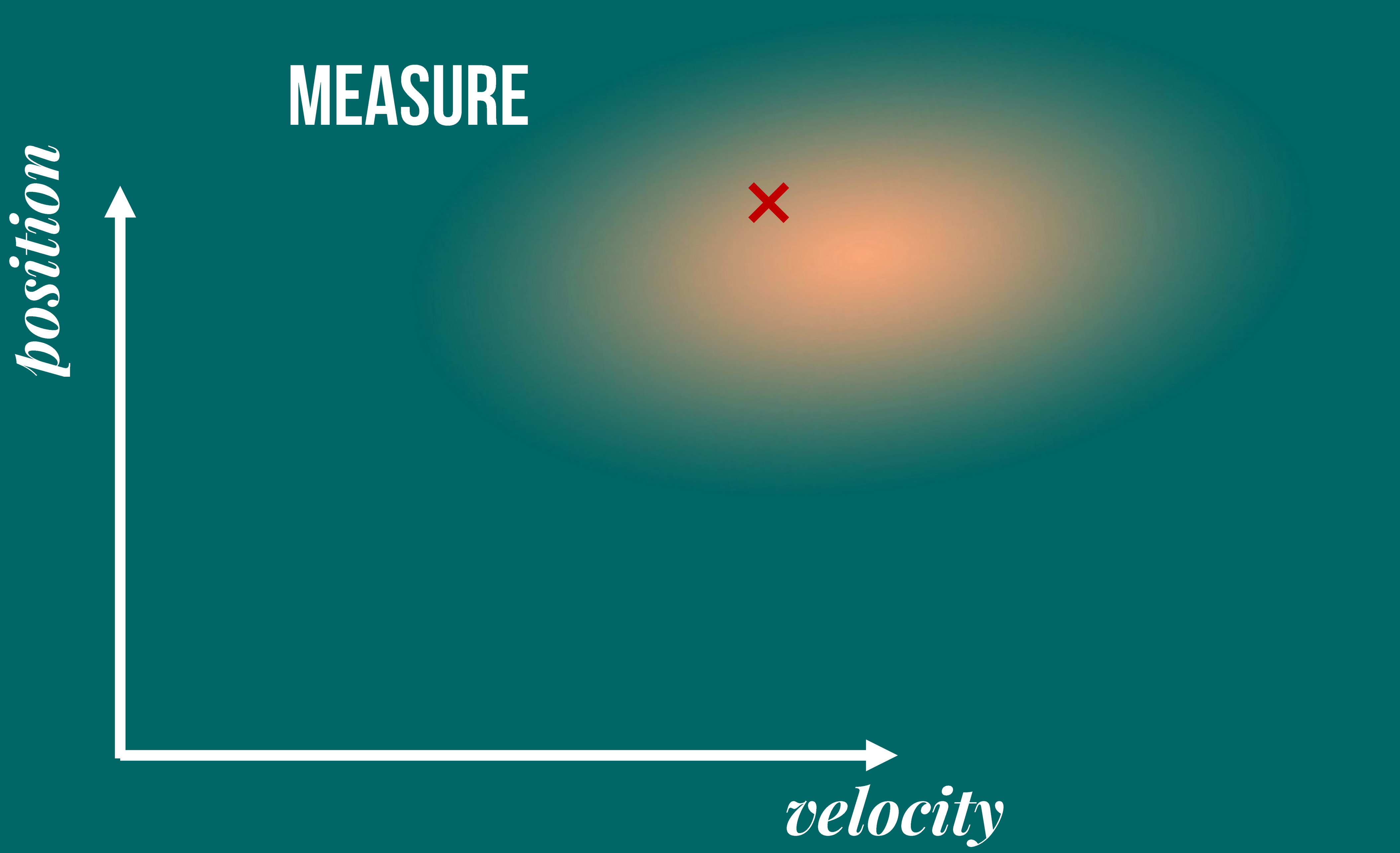
Extrapolate the state:

$$x_i^{predicted} = \phi_{i-1}\, x_{i-1}^{corrected}$$

Extrapolate the state error covariance:

$$P_i^{predicted} = \phi_{i-1}\, P_{i-1}^{corrected}\, \phi_{i-1}^{T} + Q_{i-1}$$

MEASURE

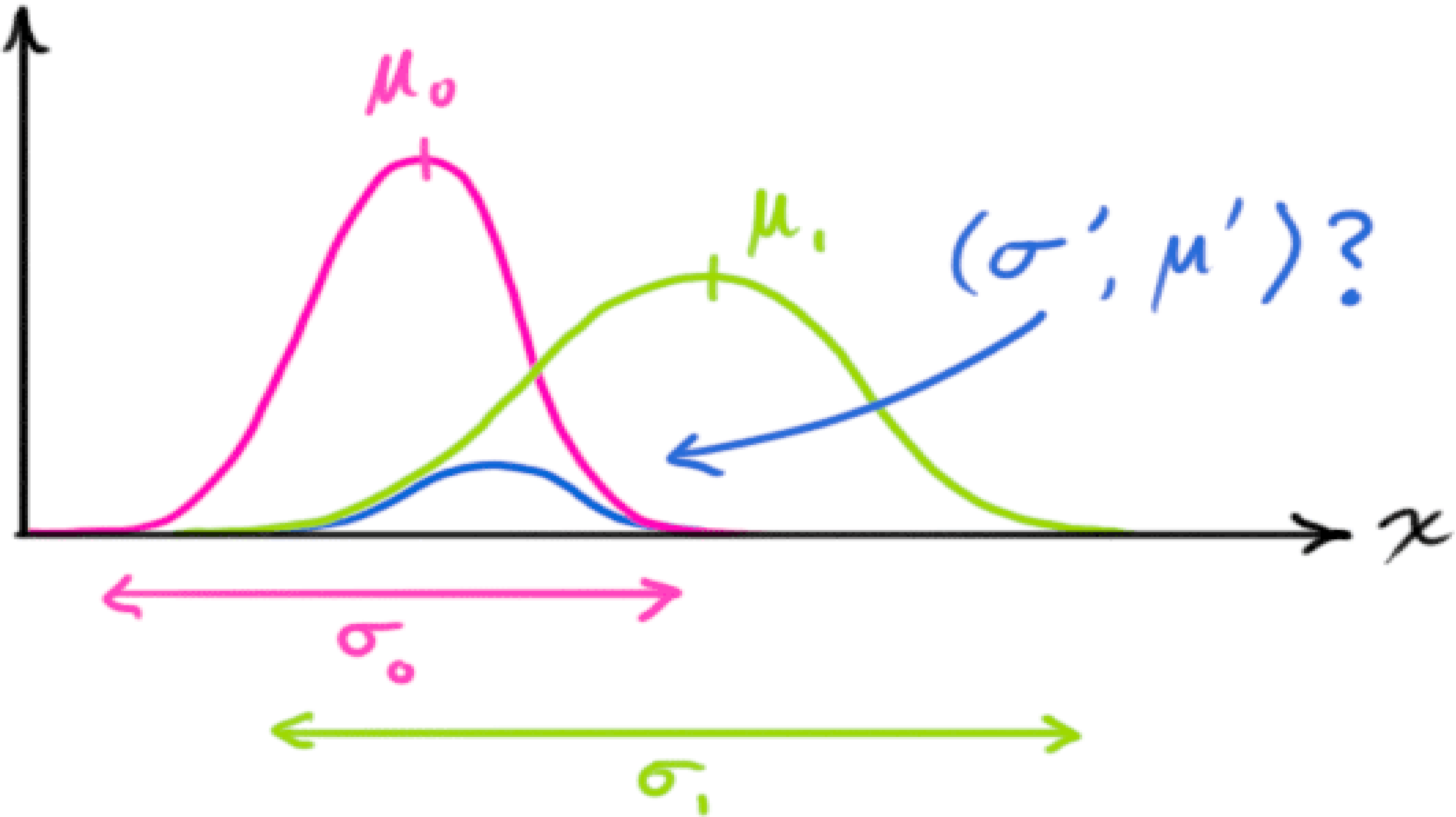


MEASURE

STEP 2: MEASURE

Compute the Kalman gain:

$$K_i = \frac{P_i^{predicted} H_i^T}{H_i P_i^{predicted} H_i^T + R_i}$$



μ₀
μ₁
(σ′; μ′)?
x
σ₀
σ₁

$$\mu' = \mu_0 + \frac{\sigma_0^2(\mu_1 - \mu_0)}{\sigma_0^2 + \sigma_1^2}$$

$$\boldsymbol{\sigma}'^2 = \sigma_0^2 - \frac{\sigma_0^4}{\sigma_0^2 + \sigma_1^2}$$

$$\mathbf{k} = \frac{\sigma_0^2}{\sigma_0^2 + \sigma_1^2}$$

$$\mu' = \mu_0 + \mathbf{k}(\mu_1 - \mu_0)$$

$$\boldsymbol{\sigma}'^2 = \sigma_0^2 - \mathbf{k}\sigma_0^2$$

STEP 2: MEASURE

Update extrapolations with new measurements:

$$x_i^{corrected} = x_i^{predicted} + K_i\left(z_i - H_i x_i^{predicted}\right)$$

$$P_i^{corrected} = (I - K_i H_i) P_i^{predicted}$$

$$\mathbf{k} = \frac{\sigma_0^2}{\sigma_0^2 + \sigma_1^2}$$

$$\mu' = \mu_0 + \mathbf{k}(\mu_1 - \mu_0)$$

$$\sigma'^2 = \sigma_0^2 - \mathbf{k}\sigma_0^2$$

VISUAL TUTORIAL

http://www.bzarg.com/p/how-a-kalman-filter-works-in-pictures/

Multi Activity Recognition Based on Bodymodel-Derived Primitives

Andreas Zinnen[1,2], Christian Wojek[2], and Bernt Schiele[2]

[1] SAP Research, CEC Darmstadt, Germany
[2] Computer Science Department, TU Darmstadt, Germany
{andreas.zinnen,wojek,schiele}@cs.tu-darmstadt.de

Abstract. We propose a novel model-based approach to activity recognition using high-level primitives that are derived from a human body model estimated from sensor data. Using short but fixed positions of the hands and turning points of hand movements, a continuous data stream is segmented in short segments of interest. Within these segments, joint boosting enables the automatic discovery of important and distinctive features ranging from motion over posture to location. To demonstrate the feasibility of our approach we present the user-dependent and across-user results of a study with 8 participants. The specific scenario that we study is composed of 20 activities in quality inspection of a car production process.

Keywords: Activity Recognition, Boosting, Human-Body Model.

RECAP

Sensor Selection

Feature Selection

Kalman Filter

Reporting Activity Recognition Results

Research Methods for Identifying Relevant Activities