15-494/694: Cognitive Robotics

Dave Touretzky

Lecture 13:

More on Particle Filters

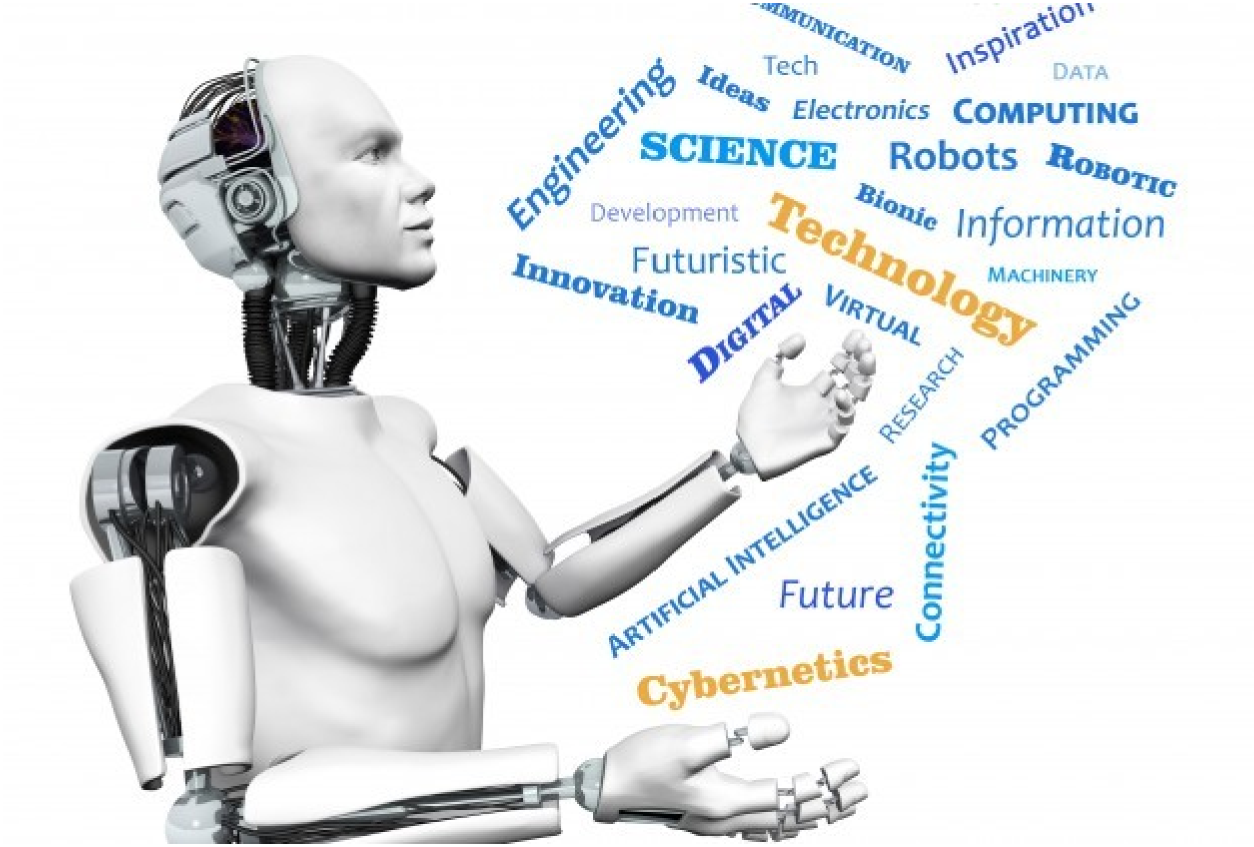


Image from http://www.futuristgerd.com/2015/09/10

Outline

- Error ellipses
- Particle SLAM

Particles Represent a Distribution

- Particles provide a non-parameteric representation for a distribution.
- Advantage: can represent distributions that are multi-modal.
- How do we display the “variance” in such a distribution? Pretend its gaussian.

Parametric Distribution

- Gaussian distribution has a mean μ and variance σ^2.
- For planar pose we have a mean vector and a covariance matrix:

$$\vec{\mu} = \begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix} \qquad \Sigma = \begin{bmatrix} \sigma_{xx} & \sigma_{xy} \\ \sigma_{xy} & \sigma_{yy} \end{bmatrix}$$

- Separately we have a heading θ and a variance for θ.

Mean and Covariance

$$W = \sum_i w_i$$

$$\vec{\mu} = \frac{1}{W} \sum_i \begin{bmatrix} x_i \\ y_i \end{bmatrix} w_i$$

$$\Sigma = \frac{1}{W} \sum_i \begin{bmatrix} x_i - \mu_x \\ y_i - \mu_y \end{bmatrix} \cdot \begin{bmatrix} x_i - \mu_x \\ y_i - \mu_y \end{bmatrix}^T w_i = \begin{bmatrix} \sigma_{xx} & \sigma_{xy} \\ \sigma_{xy} & \sigma_{yy} \end{bmatrix}$$

Direction of the Error Ellipse

- Fine the eigenvectors and eigenvalues of the covariance matrix Σ.

- The eigenvectors give the orientation of the major and minor axes.
  - Python: can use np.linalg.eigh(sigma) because sigma is symmetric.

- The eigenvalues give the squares of the lengths of the axes.

Heading Variance

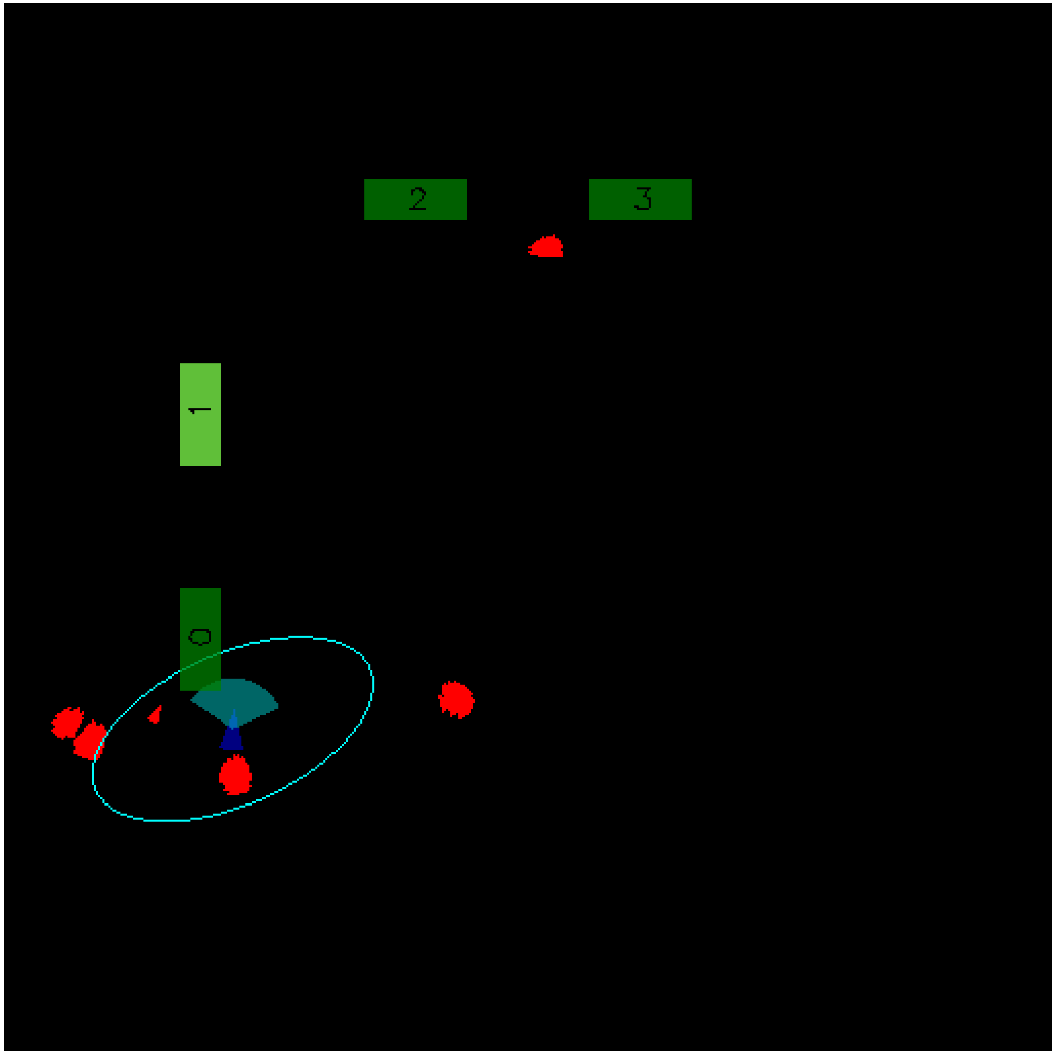
Heading is a circular variable, so we use different statistics.

$$\mu_\theta = \arctan\left(\frac{1}{W}\frac{\sum_i \sin(\theta_i) w_i}{\sum_i \cos(\theta_i) w_i}\right)$$

$$R_{sq} = \left(\sum_i \cos(\theta_i) w_i\right)^2 + \left(\sum_i \sin(\theta_i) w_i\right)^2$$

$$var_\theta = 1 - \sqrt{R_{sq}} / W$$

Error Ellipse

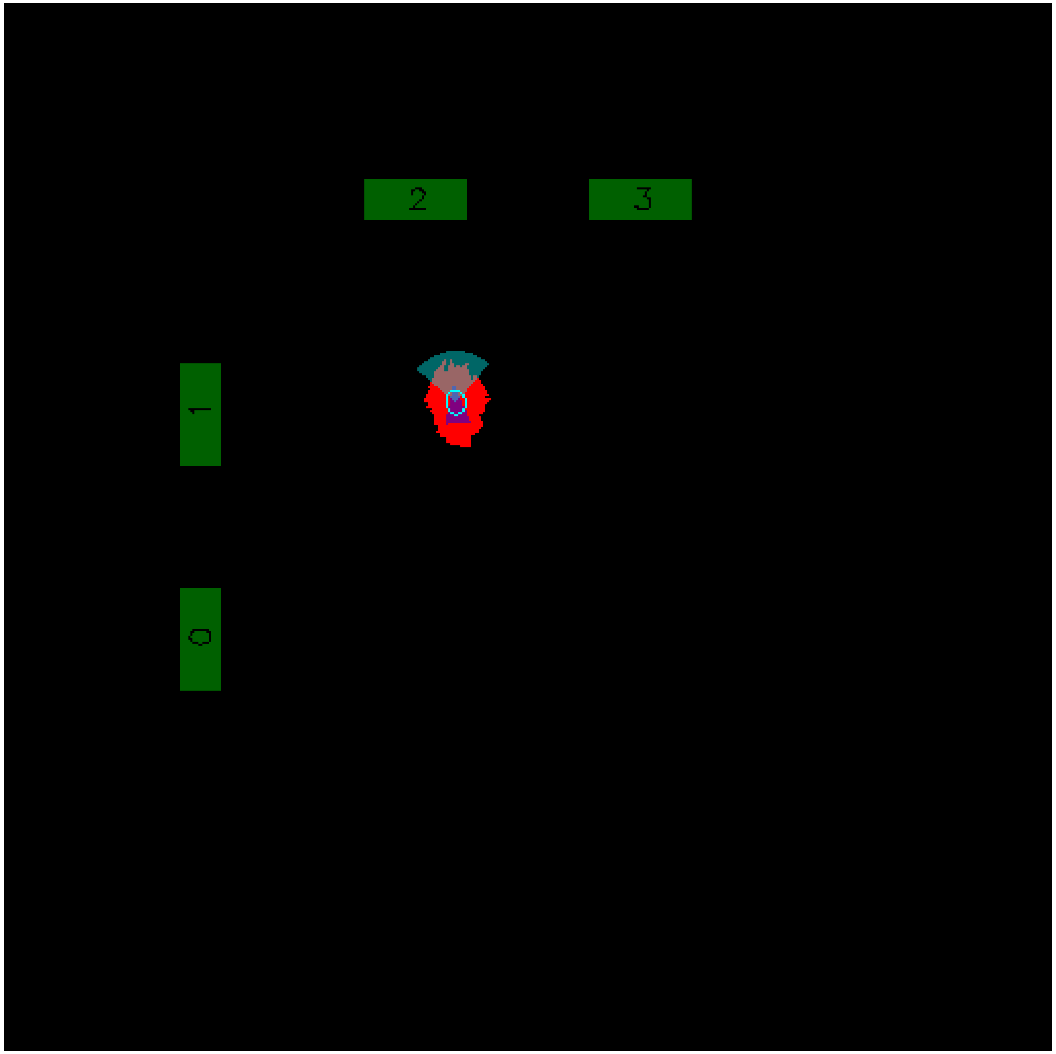


Heading Variance

- The value of the circular variance measure ranges from 0 to 1.

- Plot heading standard deviation as a wedge of angular width 0 to 360°.

Heading Variance



Particle SLAM

- Infer the map at the same time as we capture our position.
- Each particle i contains:
  - Robot pose estimate x_i, y_i, θ_i
  - Kalman filter for each landmark j:
    - $\mu_x^{(j)}$, $\mu_y^{(j)}$
    - $\Sigma^{(j)}$
- New landmarks are added to the set as they are encountered.