TU München
Fakultät für Informatik
PD Dr. Rudolph Triebel
John Chiotellis

Machine Learning for Computer Vision
Summer term 2017

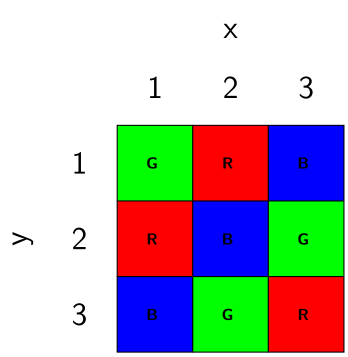
July 15, 2017
Topic: Hidden Markov Models and Sampling methods

Exercise 1: Viterbi Algorithm

We play again with our robot from the first homework assignment. As we mentioned back then the robot has a camera with an observation model that looks as follows:

Sensed color \ Actual color	R	G	B
R	0.8	0.1	0.1
G	0.1	0.6	0.2
B	0.1	0.3	0.7

This time we put the robot in a room where the floor looks like this:



a) What is the state space? What is the observation space? Draw the trellis diagram.

b) Assume the robot can only move vertically and horizontally. We let the robot move randomly. If the attempted move leads outside of the bounds of the room the robot stays at its current position. Compute the state transition matrix.

c) After 3 time steps, what is most likely the path that the robot followed if the camera reads $\{z_1 =$ R, $z_2 =$G, $z_3 =$ G $\}$? Assume the robot's initial position is unknown.

Exercise 2: Kullback-Leibler divergence

a) What does the KL divergence describe? Which are its key properties?

b) Compute the KL-divergence of two univariate normal distributions.
What if they have the same mean? What if they have the same variance?

c) Consider a factorized variational distribution $q(Z)$. By using the technique of Lagrange multipliers, verify that minimization of $KL(p||q)$ with respect to one of the factors $q_i(Z_i)$ keeping all other factors fixed, leads to the solution:

$$q_j^*(Z_j) = \int p(Z) \prod_{i \neq j} dZ_i = p(Z_j)$$

Exercise 3: Particle Filter

Theory:

a) What kind of spaces can we explore with a particle filter?

b) What kind of distributions can we approximate with a particle filter?

c) In a Monte Carlo localization problem what do the particles and the particle weights represent?

Programming : Implementing a particle filter for global localization.
Assume we have a robot in a 2D world of size 100m x 100m. The world is cyclic so if the robot crosses some border it ends up on the opposite side. There are four landmarks in this world at positions (20,20), (80,20), (20,80) and (80,80). The robot can measure its distance to each of the landmarks through its sensor and thus it estimates its true position using a measurement model with $\sigma_{sense} = 3$. This time our robot's state is described by 3 variables, x, y and θ (orientation). The robot's motion model consists of noise $\sigma_{tra} = 0.1$ for the translational motion and $\sigma_{rot} = 0.05$ for the rotational motion.

- Implement a function *move* that takes as arguments a turning angle and a forward motion distance.
- Implement a function *sense* that computes the estimated distance to the landmarks.
- Implement the function *measurement_prob* that computes the likelihood of a sensor measurement.
- Implement the function *resample* using the low variance method from the lecture.

Initialize the robot randomly and use a particle set of 1000 particles. Compute the mean error (between particles and true state). Let the robot move, turning by 0.1 radians and moving 5 meters forward. Compute the new weights and resample. Compute the mean error again. Iterate.

Exercise 4: Gibbs sampling

Show that the Gibbs sampling algorithm satisfies detailed balance:

$$p^*(z)T(z, z') = p^*(z')T(z', z)$$