Machine Learning for Probabilistic Robotics with Webots

Joan Gerard[1]
Promotor: Prof. Gianluca Bontempi [1]

[1]Université Libre de Bruxelles

March 4, 2020

Table of Contents

Neural Network Model: Coordinates → Sensor

Neural Network Model: Sensors → Coordinates

Particles Filter: Weighted Average

Combination of models

Results

Table of Contents

Configuration

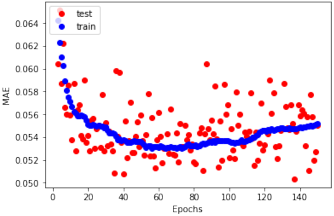
- 8 neural networks used
- Each neural network is trained using the (x, y, θ) coordinates as input and one sensor measurement as output for each model: $s_1, s_2, ..., s_8$.
- Many NN architectures were tested. The one that gave best results was:
  - Input Layer: 3 neurons
  - Intermediate Layer: Fully connected with 10 neurons
  - Intermediate Layer: Fully connected with 6 neurons
  - Intermediate Layer: Fully connected with 3 neurons
  - Output Layer: 1 neuron

Configuration

- Dropdown intermediate layers with 0.2, 0.3 and 0.5 were added.
- L2, L1 weight regularization were also added with learning rate of 0.1, 0.01 and 0.001.
- Both: Dropdown and L2, L1 weight regularization performed worst than the previously described model: the MAE decreased suddenly from first epoch to second and then it remained constant.
- Optimizer: rmsprop
- Activation: relu
- Metrics: MAE
- Loss: MSE

Results

- 150 epochs
- 5-fold
- Final model was taken with 75 epochs
- For sensor 1:



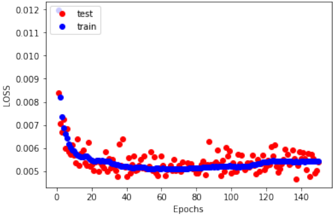


Table of Contents

Neural Network Model: Coordinates → Sensor

Neural Network Model: Sensors → Coordinates

Particles Filter: Weighted Average

Combination of models

Results

Configuration

- A neural network model that predicts the (x, y, θ) coordinates given the sensor measurements as input is created to be combined together with the previously described models.
- The neural network architecture is as follows:
  - Input Layer: 8 neurons (the sensors measurements)
  - Intermediate Layer: Fully connected with 10 neurons
  - Output Layer: 3 neurons (the coordinates)
- Optimizer: rmsprop
- Activation: relu
- Metrics: MAE
- Loss: MSE

Results

- 150 epochs
- 5-fold
- Final model was taken with 20 epochs

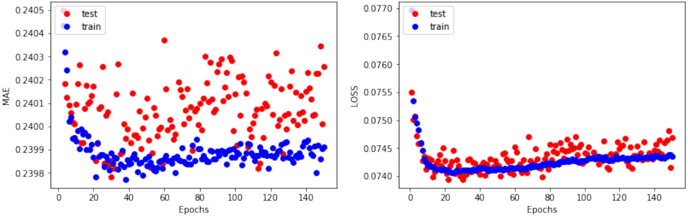


Table of Contents

Neural Network Model: Coordinates → Sensor

Neural Network Model: Sensors → Coordinates

Particles Filter: Weighted Average

Combination of models

Results

- The weighted average is taken among all the particles to select the corrected robot state.

$$\hat{s}_t = \sum_{i=1}^{n} w_t^i * s_t^i$$

- Where:
  - n is the number of particles
  - $\hat{s}_t$ is the estimated state at time t
  - w_t^i is the weight at time t of particle i
  - s_t^i is the state at time t of particle i

Table of Contents

Neural Network Model: Coordinates → Sensor

Neural Network Model: Sensors → Coordinates

Particles Filter: Weighted Average

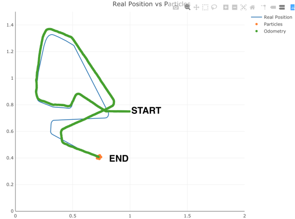
Combination of models

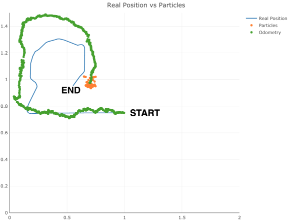
Results

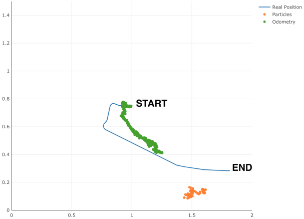
- Let $\hat{s_t}$ be the prediction state given by the particles filter.
- Let $\hat{s'_t}$ be the prediction state given by the second model (Sesors $\rightarrow$ Coordinates).
- The final model was calculated averaging both: $\frac{\hat{s_t}+\hat{s'_t}}{2}$

Table of Contents

- Right: Particles filter only. $n = 1000; \sigma_{x,y} = 0.00045; \sigma_\theta = 2.5$
- Center: Particles filter only. $n = 100; \sigma_{x,y} = 0.0045; \sigma_\theta = 2.5$
- Left: Particles filter + Second Model. $n = 100; \sigma_{x,y} = 0.0045; \sigma_\theta = 2.5$


Real Position vs Particles
Real Position
Particles
Odometry
START
END


Real Position vs Particles
Real Position
Particles
Odometry
START
END


Real Position
Particles
Odometry
START
END