RBE 550: Motion Planning

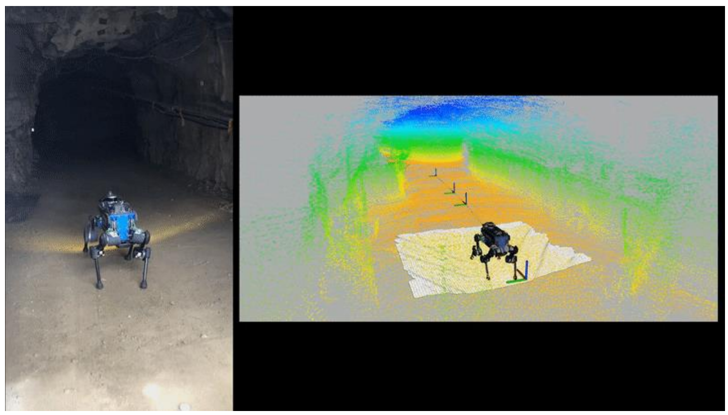
Paper Review

Gaussian Process Occupancy Maps

Authors: S. Callaghan and F. Ramos

Presenter: Sapan Agrawal

ANYmal Robot at DARPA SubT STIX

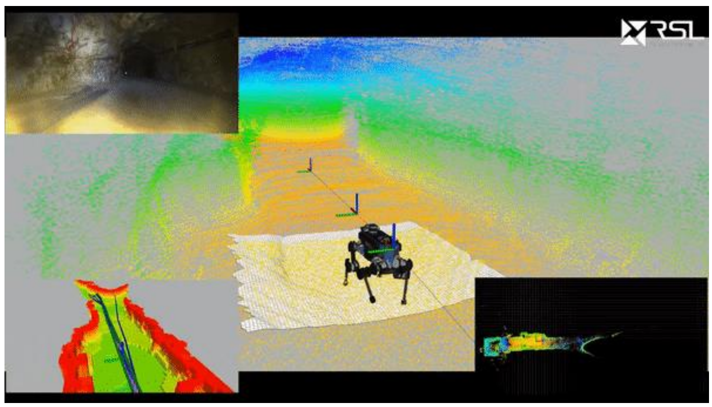


Source: https://www.youtube.com/watch?v=6eB-2Rzojh4

- This is the Darpa Subterranean Challenge where the robot has to map and navigate in underground environment.
- The robot has to plan a feasible path from it current position to the goal while avoiding the obstacles.
- perceive the environment using various sensors: lidar range sensor, camera or depth sensor, generally in form of point cloud as we see in the video.
- With this information about the environment, we want to know the location of the obstacles and where the robot can move safely.

What is Occupancy Map?

A representation map of the environment as an evenly spaced Random Variable each representing presence of obstacle in that location.



- So, this representation map which classifies the region as either occupied or freespace is known as the Occupancy Map.
- As we see in the video the occupancy map is created on the left side using the depth information from lidar sensor. The walls are classified as obstacles in the red while the free space is in green.

Problem with Occupancy grid?

What’s the problem in using the Occupancy grid map?

Problem with Occupancy grid?

- Independence relationship between the cells

What's the problem in using the Occupancy grid map?

1) Probability of cell being occupied depends only on laser beam and not influenced by neighboring cells. However, we know that due to physical structure of the objects spatial dependency exists.

Problem with Occupancy grid?

- Independence relationship between the cells
  - Inaccurate results in regions of uncertainty (occluded region, sensor noise)

What’s the problem in using the Occupancy grid map?

1) Probability of cell being occupied depends only on laser beam and not influenced by neighboring cells. However, we know that due to physical structure of the objects spatial dependency exists.
2) Other drawbacks of using traditional Occupancy map are Discretization Error and Large Memory .
    1) If discretization is very coarse, it may misrepresent a free space as occupied.
    2) On the other hand, if the discretization in fine, it requires more memory and computation time.

Problem with Occupancy grid?

- Independence relationship between the cells
  - Inaccurate results in regions of uncertainty (occluded region, sensor noise)

- Discretization Error

Second is the discretization error.
If discretization is very coarse, it may misrepresent a free space as occupied.

Problem with Occupancy grid?

- Independence relationship between the cells
  - Inaccurate results in regions of uncertainty (occluded region, sensor noise)
- Discretization Error
  - Coarse discretization can misrepresent a cell

Second is the discretization error.
If discretization is very coarse, it may misrepresent a free space as occupied.

Problem with Occupancy grid?

- Independence relationship between the cells
  - Inaccurate results in regions of uncertainty (occluded region, sensor noise)
- Discretization Error
  - Coarse discretization can misrepresent a cell
- Large memory requirement

On the other hand, if the discretization in fine, it requires more memory and computation time.

Problem with Occupancy grid?

- Independence relationship between the cells
  - Inaccurate results in regions of uncertainty (occluded region, sensor noise)
- Discretization Error
  - Coarse discretization can misrepresent a cell
- Large memory requirement
  - Fine discretization will be computationally expensive and require more memory.

On the other hand, if the discretization in fine, it requires more memory and computation time.

Why Gaussian Process?

- Uses continuous, non-linear, non-parametric function to represent map

1) Learn the relationship between the neighboring cells. So, if the sensor data is noisy, it classifies the cells correctly due to dependency.
    1) Non- parametric means there is no explicit input-output function.

Why Gaussian Process?

- Uses continuous, non-linear, non-parametric function to represent map
  - predict dependency between the neighboring cells.

1) Learn the relationship between the neighboring cells. So, if the sensor data is noisy, it classifies the cells correctly due to dependency.
    1) Non- parametric means there is no explicit input-output function.

Why Gaussian Process?

- Uses continuous, non-linear, non-parametric function to represent map
  - predict dependency between the neighboring cells.
  - Robust to noise and sparse data

2) Any-time Algorithm: algorithms that can return a valid solution to a problem even if it is interrupted before it ends. E.g. Newton's Gradient descent method. Thus, course resolution maps can be generated quickly to generate the path.

Why Gaussian Process?

- Uses continuous, non-linear, non-parametric function to represent map
  - predict dependency between the neighboring cells.
  - Robust to noise and sparse data

- Any-time Algorithm

2) Any-time Algorithm: algorithms that can return a valid solution to a problem even if it is interrupted before it ends. E.g. Newton's Gradient descent method. Thus, course resolution maps can be generated quickly to generate the path.

Why Gaussian Process?

- Uses continuous, non-linear, non-parametric function to represent map
  - predict dependency between the neighboring cells.
  - Robust to noise and sparse data

- Any-time Algorithm
  - Can get both coarse and fine maps.

2) Any-time Algorithm: algorithms that can return a valid solution to a problem even if it is interrupted before it ends. E.g. Newton's Gradient descent method. Thus, course resolution maps can be generated quickly to generate the path.

Why Gaussian Process?

- Uses continuous, non-linear, non-parametric function to represent map
  - predict dependency between the neighboring cells.
  - Robust to noise and sparse data

- Any-time Algorithm
  - Can get both coarse and fine maps.

- Variance plot can be used to highlight unexplored region

1) Learn the relationship between the neighboring cells. So, if the sensor data is noisy, it classifies the cells correctly due to dependency.
    1) Non- parametric means there is no explicit input-output function.

2) Any-time Algorithm: algorithms that can return a valid solution to a problem even if it is interrupted before it ends. E.g. Newton's Gradient descent method. Thus, course resolution maps can be generated quickly to generate the path.

3) Additional feature is that variance can be used to find the unexplored regions.

Why Gaussian Process?

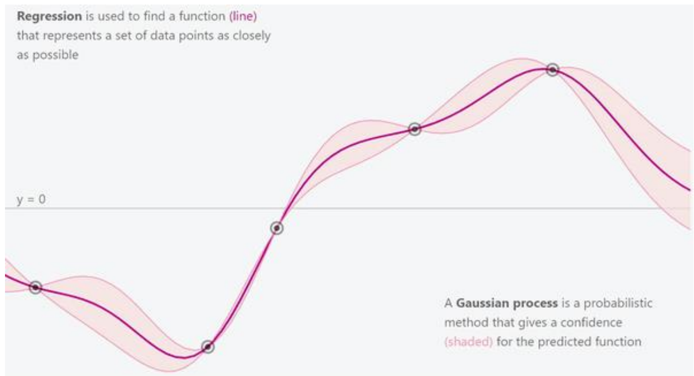
- Uses continuous, non-linear, non-parametric function to represent map
  - predict dependency between the neighboring cells.
  - Robust to noise and sparse data
- Any-time Algorithm
  - Can get both coarse and fine maps.
- Variance plot can be used to highlight unexplored region
  - Additional benefit

1) Learn the relationship between the neighboring cells. So, if the sensor data is noisy, it classifies the cells correctly due to dependency.
    1) Non- parametric means there is no explicit input-output function.

2) Any-time Algorithm: algorithms that can return a valid solution to a problem even if it is interrupted before it ends. E.g. Newton's Gradient descent method. Thus, course resolution maps can be generated quickly to generate the path.

3) Additional feature is that variance can be used to find the unexplored regions.

What is Gaussian Process?



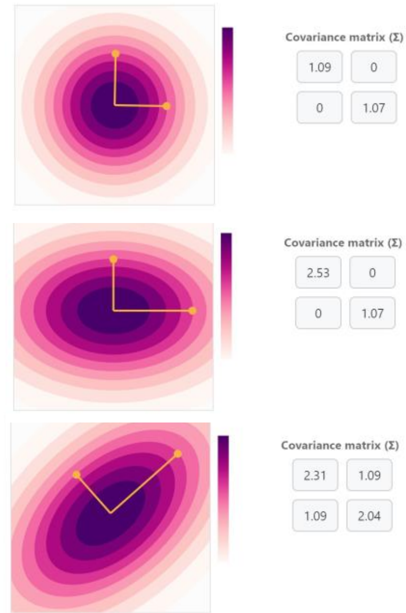
Source: https://distill.pub/2019/visual-exploration-gaussian-processes/

- Gaussian Process is a tool in Machine Learning that allows us to make predictions based on prior observed data.
- So it can be used for Fitting a function (regression problem), classification and clustering.
- For a given training points there are infinitely many functions that can fit the data.
- GP assigns the probability to these functions.

Gaussian Process

$$P_{X,Y} = \begin{bmatrix} X \\ Y \end{bmatrix} \sim \mathcal{N}(\mu, \Sigma) = \mathcal{N}\left(\begin{bmatrix} \mu_X \\ \mu_Y \end{bmatrix}, \begin{bmatrix} \Sigma_{XX} \, \Sigma_{XY} \\ \Sigma_{YX} \, \Sigma_{YY} \end{bmatrix} \right)$$

With X and Y representing subsets of original random variables.

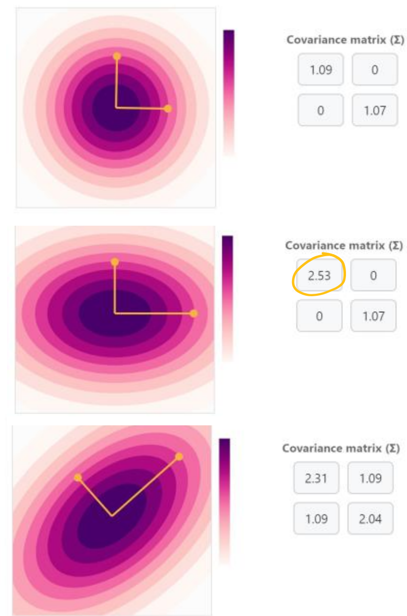


- So a gaussian distribution is defined by its mean and covariance.
- For a 2 dim gaussian distribution the covariance is a 2x2 matrix.
- In the figures we see how the covariance matrix affect the shape of the gaussian distribution.

Gaussian Process

$$P_{X,Y} = \begin{bmatrix} X \\ Y \end{bmatrix} \sim \mathcal{N}(\mu, \Sigma) = \mathcal{N}\left(\begin{bmatrix} \mu_X \\ \mu_Y \end{bmatrix}, \begin{bmatrix} \Sigma_{XX} & \Sigma_{XY} \\ \Sigma_{YX} & \Sigma_{YY} \end{bmatrix} \right)$$

With X and Y representing subsets of original random variables.

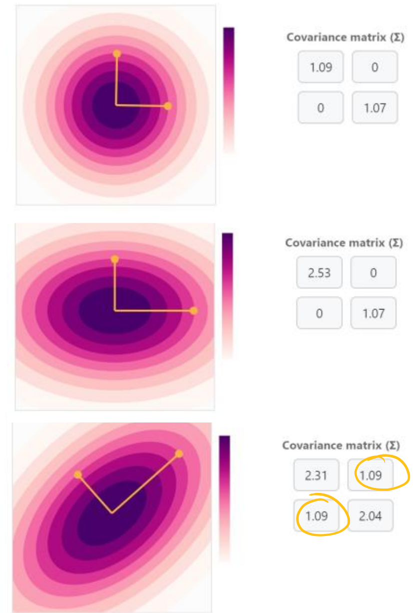


- Higher value of XX position of Covariance matrix means higher variance in the X direction.

Gaussian Process

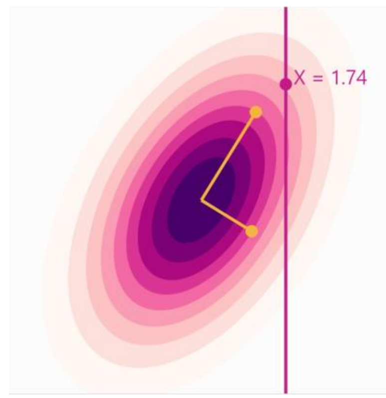
$$P_{X,Y} = \begin{bmatrix} X \\ Y \end{bmatrix} \sim \mathcal{N}(\mu, \Sigma) = \mathcal{N}\left(\begin{bmatrix} \mu_X \\ \mu_Y \end{bmatrix}, \begin{bmatrix} \Sigma_{XX} \, \Sigma_{XY} \\ \Sigma_{YX} \, \Sigma_{YY} \end{bmatrix} \right)$$

With X and Y representing subsets of original random variables.



- The XY position in the Covariance matrix (in figure 3) tells us how strongly the X and Y random variables are corelated to each other.
- So in figures 1 and 2 the X and Y Random variables are independent of each other.
-

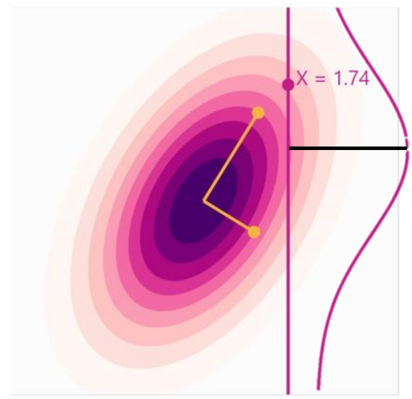
Gaussian Process



So lets go back to the corelated guassian distribution.

- So, someone asks us given precisely value of X to be 1.74 whats the value of Y?
- It looks like y can take many values.

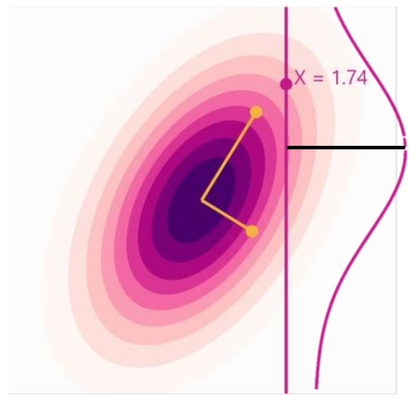
Gaussian Process



- It comes out to be Gaussian distribution.

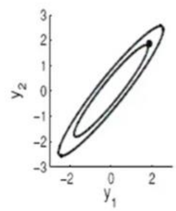
Gaussian Process

Conditioning
Gives the correlation between variables.

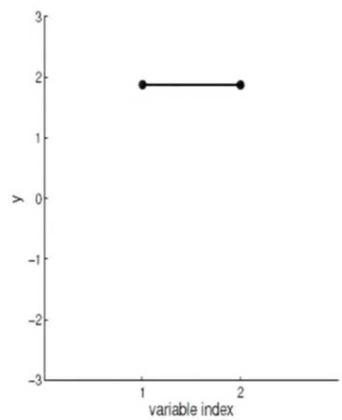


- This operation is known as conditioning.
- It basically gives us the probability of one variable depending on another variable.

Gaussian Process



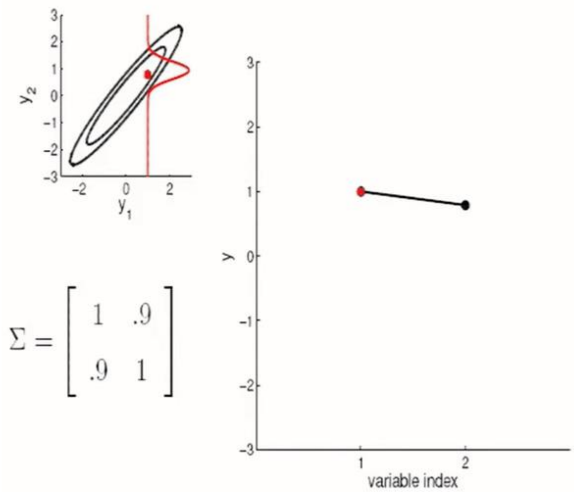
$$\Sigma = \begin{bmatrix} 1 & .9 \\ .9 & 1 \end{bmatrix}$$



Source: https://www.youtube.com/watch?v=Jv25sg-IYHU

Suppose now for the corelated gaussian distribution, I place my variables along X axis and their value along Y.
We get new representation. So here both the variables y1 and y2 have the same value of 2.

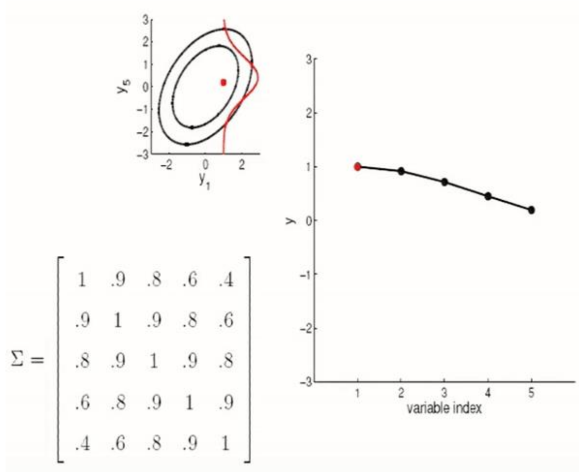
Gaussian Process



$$\Sigma = \begin{bmatrix} 1 & .9 \\ .9 & 1 \end{bmatrix}$$

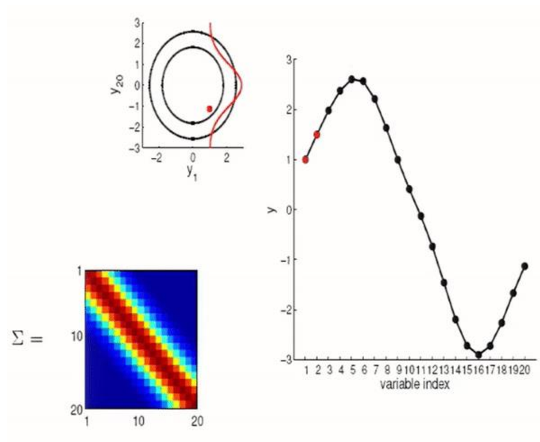
Now if we fix the variable 1 and sample variable 2 from the conditioned distribution. The line in the figure just wiggles at the pivot.

Gaussian Process



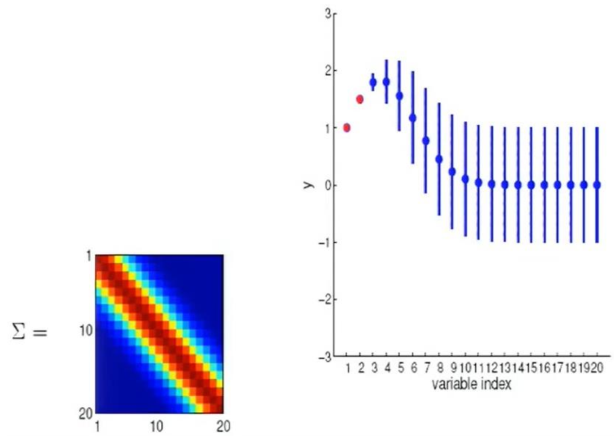
So extending to the higher dimension, we get this.

Gaussian Process



Representing the Covariance matrix in the colormap for higher dimensions.

Gaussian Process

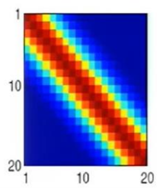


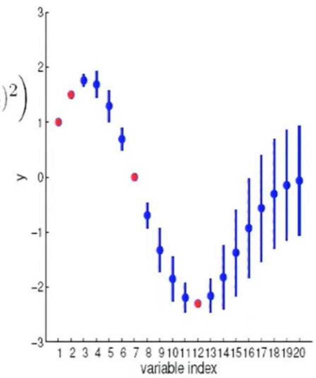
So if we average over all the random sample, we get this nice looking non-linear regression which goes through the points that we might observe. So given y1 and y2, by learning enough I can predict the values of the variables.

Gaussian Process

$$\Sigma(x_1, x_2) = K(x_1, x_2) + I\sigma_y^2$$

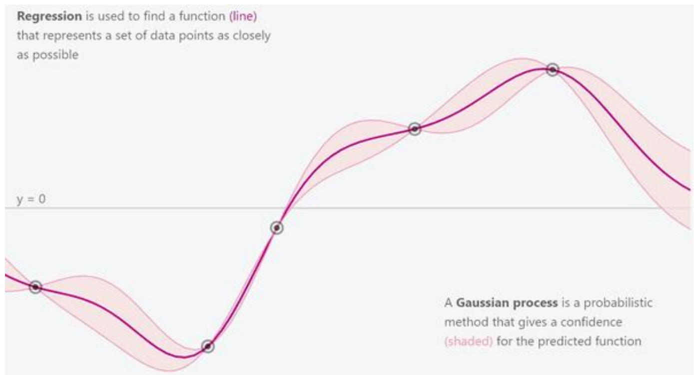
$$K(x_1, x_2) = \sigma^2 \exp\left(-\frac{1}{2l^2}(x_1 - x_2)^2\right)$$

$\Sigma =$ 



So how do we come up with this powerful covariance matrix.
The answer is the covariance function also commonly known as kernel.
So basically this linear kernel over here, has strong correlation between the variables which are close to each other.
And the dependency decreases as they go away from each other.

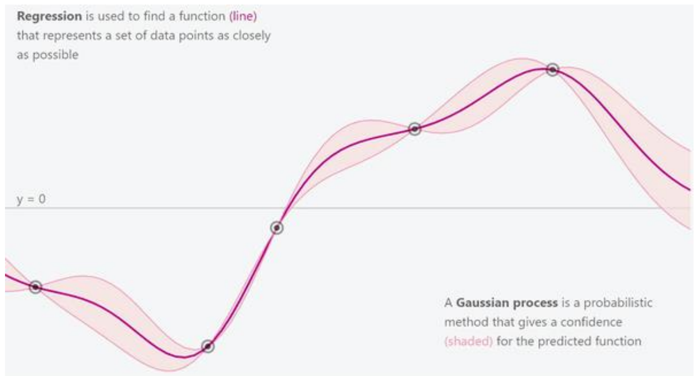
What is Gaussian Process?



Source: https://distill.pub/2019/visual-exploration-gaussian-processes/

- Gaussian Process is a tool in Machine Learning that allows us to make predictions based on prior observed data.
- So it can be used for Fitting a function (regression problem), classification and clustering.

What is Gaussian Process?

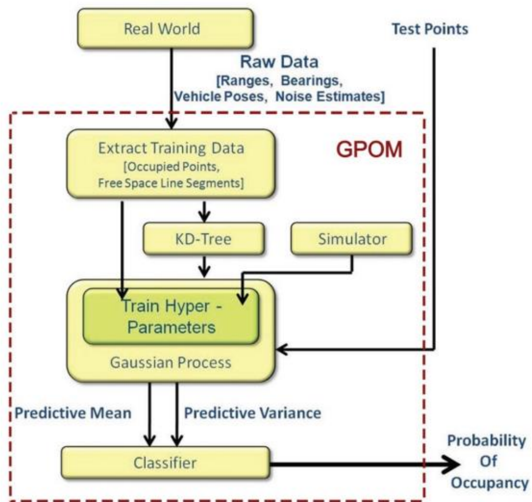


Our Problem:
To find a non-linear function that can predict the occupancy of a region given the sensor information and prior knowledge.

- So our problem is:

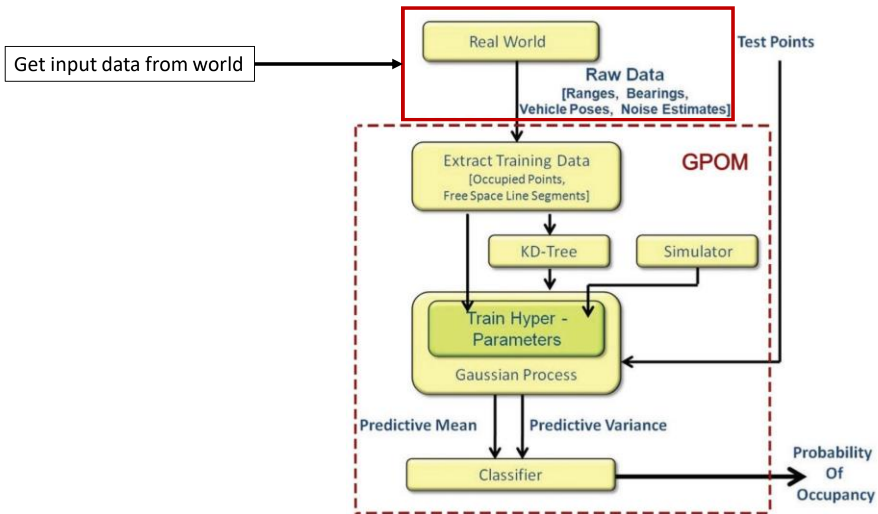
To find a non-linear function that can predict the occupancy of a region sensor information and prior knowledge i.e. location and occupancy of the training data

Algorithm

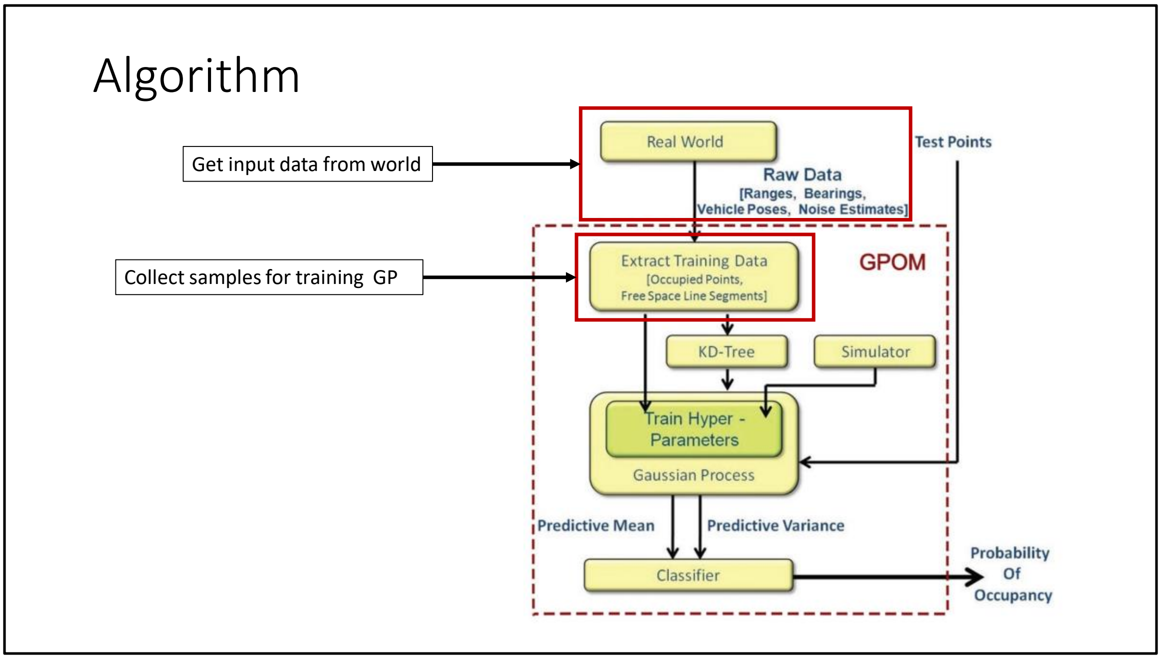


Real World
Test Points
Raw Data
[Ranges, Bearings,
Vehicle Poses, Noise Estimates]
Extract Training Data
[Occupied Points,
Free Space Line Segments]
GPOM
KD-Tree
Simulator
Train Hyper -
Parameters
Gaussian Process
Predictive Mean
Predictive Variance
Classifier
Probability
Of
Occupancy

Algorithm

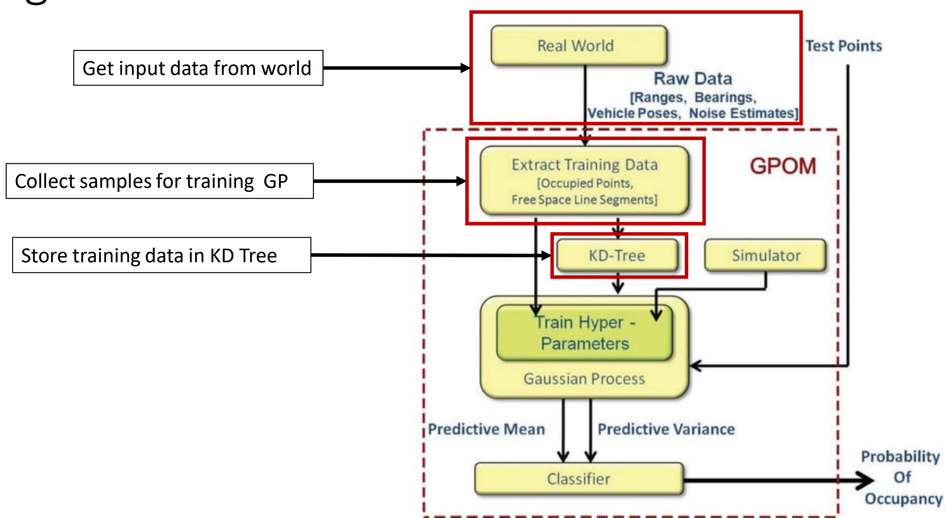


1) We get the sensor data about the environment.

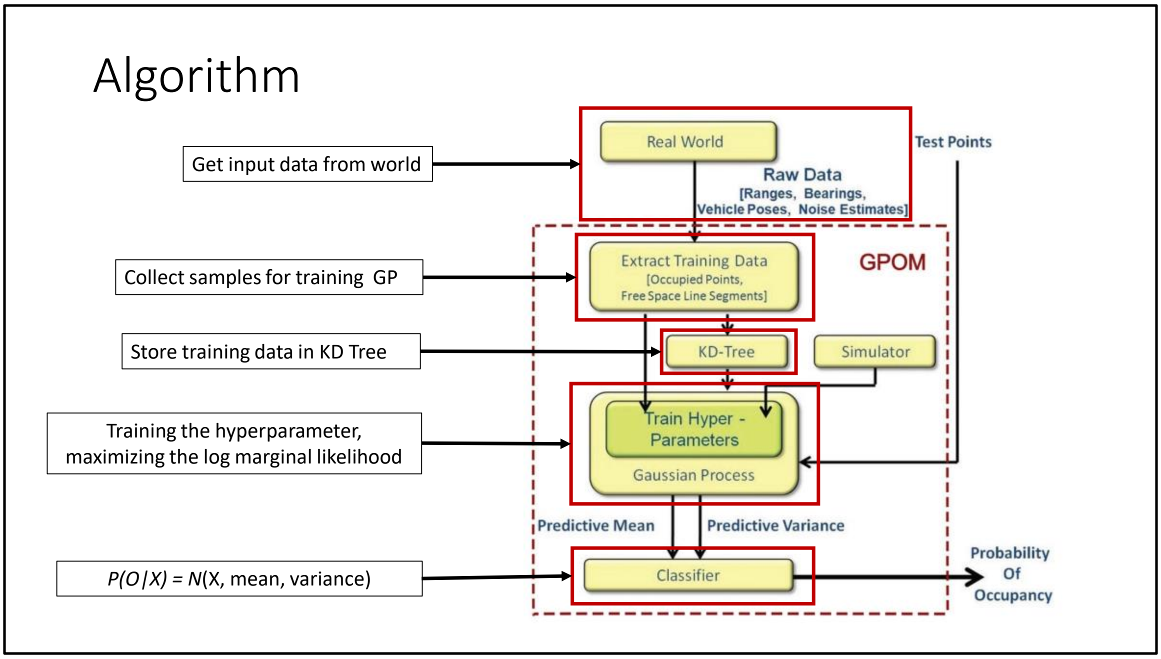


2) The next step is to collect some training samples to train a GP.

Algorithm



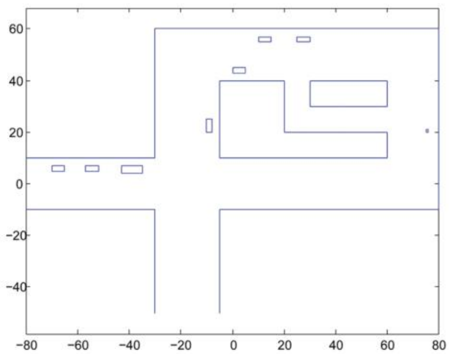
- GP requires the inversion of the covariance matrix which has O(n^3) computation complexity. n is the number of training samples.
- As points which are close have greater influence on the covariance function compared to those at distance, we can get samples near the test points and make smaller invertible covariance matrix.
- This process is done for the training samples.



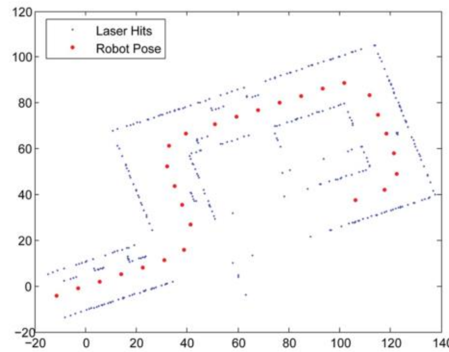
So now we get the probability of a point in the region to be classified as free space or occupied.

Overall, it’s a online classification learning problem where we locally approximate the function.

Results



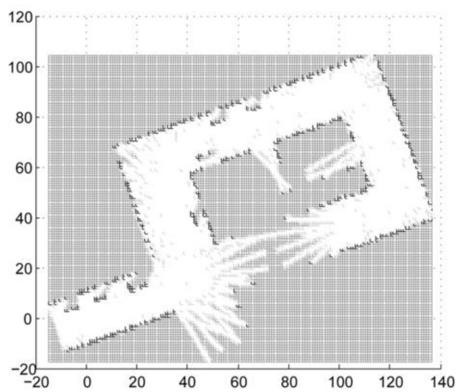
(a)
Ground Truth



(b)
Laser Return and robot Pose

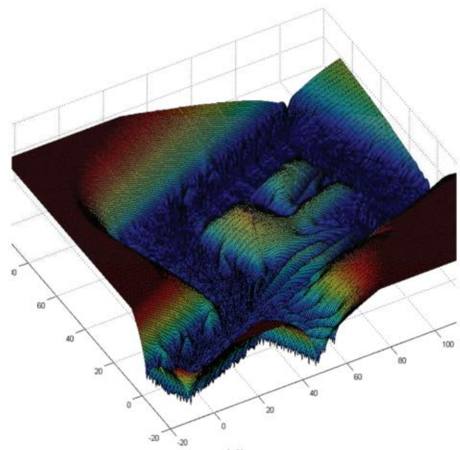
(a) Ground Truth
(b) Laser Return and Robot Pose

Results



(c)

Occupancy Grip

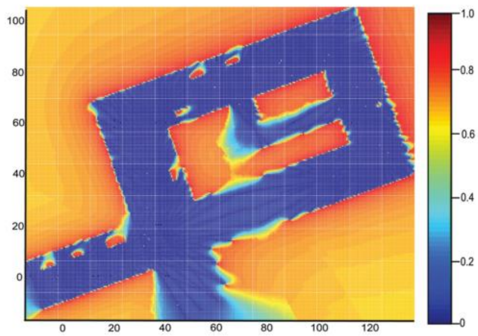


(d)

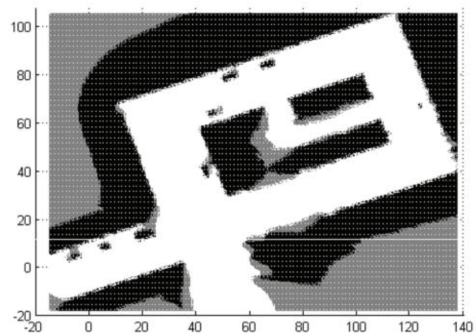
Predicted variance of GP

(c) Occupancy Grip
(d) Predicted variance of GP (interestingly very close to repulsion map studied in class)

Results



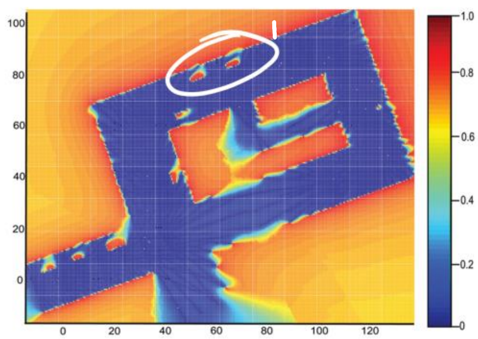
(e)
probability of occupancy versus location using the GP approach



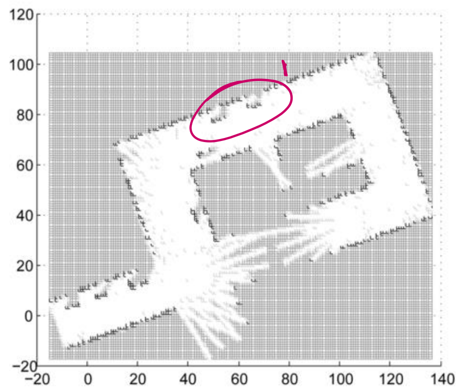
(f)
classified GPOM
Black = Occupied (p(O | x) ≥ 0.65)
Grey = Unsure (0.3 < p(O | x) < 0.7)
White = Free space (p(O | x) ≤ 0.35)

(e)This is the result that we get after training the hyperparameters. It shows the probability of a space being occupied.
(f) The figure on the right is the final output of the classification process. Note the grey region is the area that the robot can explore.

Results



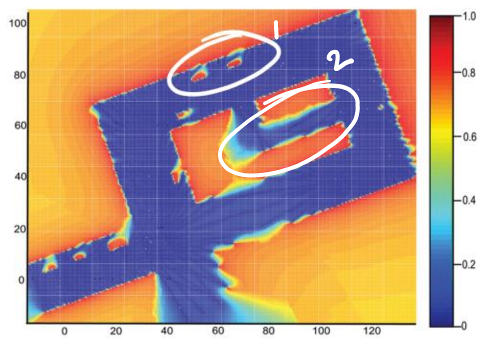
(e)
probability of occupancy versus location using the GP approach



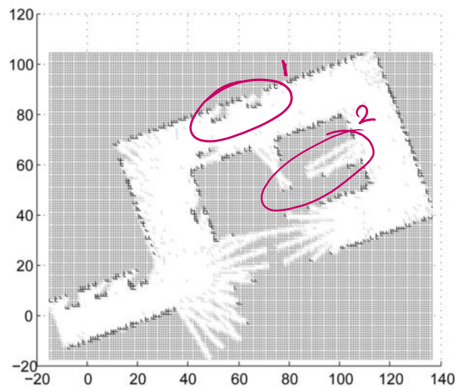
(c)
Occupancy Grip

1) Comparing the results with naïve occupancy grid map, we can see
2) in (1) how the occluded area behind the two blocks was unmarked in the occupancy grid (slide 9) is now perfectly classified due to dependency of neighboring cells.
3) Whereas in (2) the region is correctly predicted as safe path even though the sensor data was not available.

Results



(e)
probability of occupancy versus
location using the GP approach



(c)
Occupancy Grip

1) Comparing the results with naïve occupancy grid map, we can see
2) in (1) how the occluded area behind the two blocks was unmarked in the occupancy grid (slide 9) is now perfectly classified due to dependency of neighboring cells.
3) Whereas in (2) the region is correctly predicted as safe path even though the sensor data was not available.

Conclusion

- learn dependencies between the points.
  - Robust to sensor noise and occlusion
- Continuous probabilities allow 'Any-time' algorithm.
  - We can achieve arbitrary accuracy in discretization
- Provides prediction of occupancy with variance
  - Useful in identifying the unexplored regions
- Works with sparse data
  - Saves memory and computation time

- Gaussian process provide several benefits over the naïve occupancy grid map.

1) It can learn the dependencies between the points and thus it is robust to sensor noise and occlusion.
2) As we learn a continuous, non-linear function which becomes more and more accurate with time and training data, it is an anytime algorithm. We can stop the training process to any arbitrary precision.
3) The variance can be used to optimize the path planning algorithm to maximise the robot's understanding of it's environment.
4) Because we now it is a regression problem, we can predict the occupancy using few sample points, saving both memory and computation time.

References

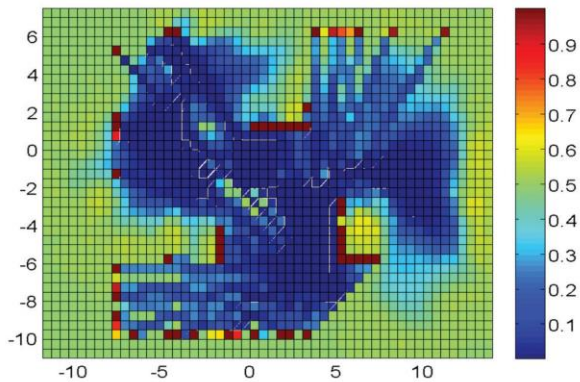
- O’Callaghan, S. T., & Ramos, F. T. (2012). Gaussian process occupancy maps. *The International Journal of Robotics Research*, *31*(1), 42-62.

- https://distill.pub/2019/visual-exploration-gaussian-processes/

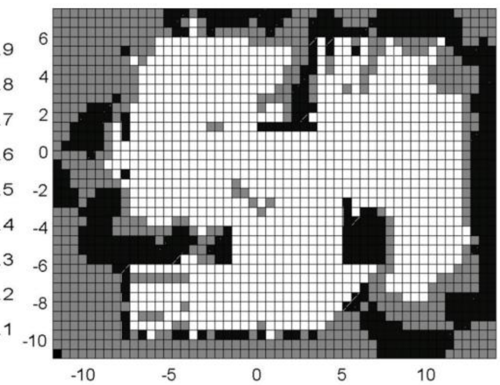
- 2 - Richard Turner, Introduction to Gaussian Processes - https://www.youtube.com/watch?v=Jv25sg-IYHU

Thank You

More Results



Occupancy Map

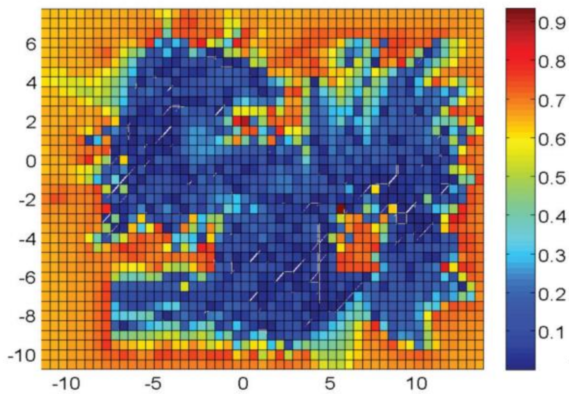


Classified Occupancy Map

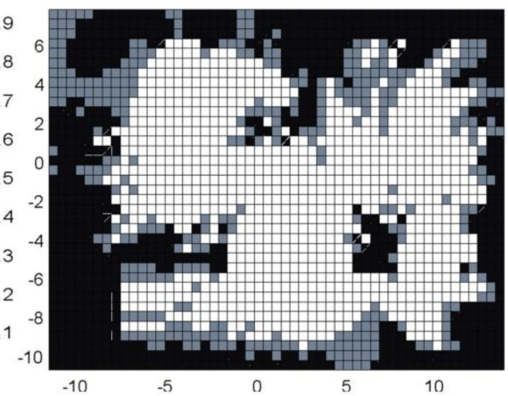
Probability of occupancy versus location prior to thresholding are shown in the left column. Reddish areas indicate regions with high probability of occupancy while bluish regions suggest the area is most likely free space. (Colour refers to the online version.)

The right column illustrates their corresponding classified maps after applying thresholds. Classification labels: Black = Occupied; White = Unoccupied; Grey = Unsure.

More Results



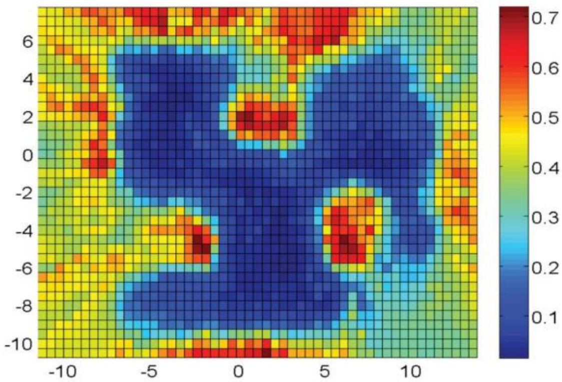
GPFOL: mapping technique that assumes negligible input noise



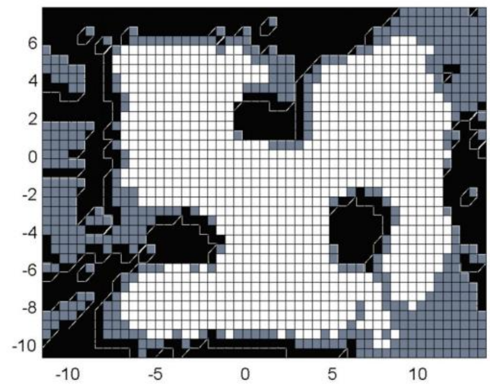
Classified GPFOL Map

GPFOL: Gaussian Process that assumes Fully observable locations

More Results



GPPOL: method which incorporates uncertainty in the observations.



Classified GPPOL Map

GPPOL: Gaussian Process that assumes partially observable locations