
HA1 Analysis

Table of Contents

1 Transformation of Gaussian random variables	1
1a	1
1b	2
2 Snow depth in Norway	2
2a	2
2b	3
2c	4
3 MMSE and MAP estimator for Gaussian mixture posteriors	4

1 Transformation of Gaussian random variables

1a

```
close all;clc;
mu = [10; 0];
cov = [0.2 0; 0 8];

A = [1 1; 1 -1];
b = [0; 0];

f = @(x) A*x;

figure(1)
for i = 50:50:150
    N = i;
    [mu_y, sigma_y] = affineGaussianTransform(mu, cov, A, b);
    [mu_y_approx, sigma_y_approx] = approxGaussianTransform(mu, cov,
f, N);

    % Observations
    level = 3;
    points = 100;
    xy = sigmaEllipse2D(mu_y, sigma_y, level, points);
    xy_approx = sigmaEllipse2D(mu_y_approx, sigma_y_approx, level,
points);

    subplot(3,1,i / 50);
    h1 = plot(xy(1,:), xy(2,:));

    axis equal
```

```

hold on
h2 = plot(xy_approx(1,:), xy_approx(2,:));

plot(mu_y(1), mu_y(2), '*', 'color', h1.Color);
plot(mu_y_approx(1), mu_y_approx(2), '*', 'color', h2.Color);
title("3\sigma level curves with N = " + N);
legend('Analytical 3\sigma', 'Approximate 3\sigma', 'Analytical
\mu', 'Approximate \mu');
end

```

1b

```

close all;

f = @(x) [norm(x); atan2(x(2),x(1))];
figure(2)
spacing = 500;
n = 1;
for i = [100, 1000, 5000]

    [mu_y, sigma_y, y] = approxGaussianTransform(mu, cov, f, i);

    % Observations
    level = 3;
    points = 100;

    xy = sigmaEllipse2D(mu_y, sigma_y, level, points);

    subplot(1,3,n);

    h1 = plot(xy(1,:), xy(2,:));
    hold on
    plot(mu_y(1), mu_y(2), '*', 'color', h1.Color);

    % Plot observed data points
    plot(y(1,:), y(2,:), '.r')

    title("N = " + i);
    legend('3\sigma', '\mu', 'Data points')
    n = n + 1;
end

```

2 Snow depth in Norway

2a

```

close all;clc;

mu_hafjell = 1.1;
mu_kvitfjell = 1;

```

```

sigma2 = 0.5^2;

y_hafjell = 1;
y_kvittfjell = 2;

sigma2_y_hafjell = 0.2^2;
sigma2_y_kvittfjell = 0.1^2;

[mu_hafjell_update, sigma_hafjell_update] = jointGaussian(mu_hafjell,
    sigma2, sigma2_y_hafjell);
xy_hafjell = sigmaEllipse2D(mu_hafjell_update, sigma_hafjell_update);

[mu_kvittfjell_update, sigma_kvittfjell_update] =
    jointGaussian(mu_kvittfjell, sigma2, sigma2_y_kvittfjell);
xy_kvittfjell = sigmaEllipse2D(mu_kvittfjell_update,
    sigma_kvittfjell_update);

figure(1)
h1 = plot(xy_hafjell(1,:), xy_hafjell(2,:));
axis equal
hold on
plot(mu_hafjell_update(1), mu_hafjell_update(2), '*', 'color',
    h1.Color);

h1 = plot(xy_kvittfjell(1,:), xy_kvittfjell(2,:));
plot(mu_kvittfjell_update(1), mu_kvittfjell_update(2), '*', 'color',
    h1.Color);

legend('3\sigma Hafjell', '\mu Hafjell', '3\sigma Kvittfjell', '\mu
    Kvittfjell')
title('Snow depth in Norway')

```

2b

```

close all; clc

% Calculate posterior gaussians.
% This can be seen as taking a slice (plane cut) out of the joint dist
% from
% previous question.
[mu_hf, sigma2_hf] = posteriorGaussian(mu_hafjell, sigma2, y_hafjell,
    sigma2_y_hafjell);
[mu_kf, sigma2_kf] = posteriorGaussian(mu_kvittfjell, sigma2,
    y_kvittfjell, sigma2_y_kvittfjell);

% Plot the result
x = 0:0.01:3;
y_hf = normpdf(x, mu_hf, sqrt(sigma2_hf));
y_kf = normpdf(x, mu_kf, sqrt(sigma2_kf));
plot(x, [y_hf; y_kf]);
xlabel('Snow depth [m]')
legend('Hafjell, \mu = ' + mu_hf, 'Kvittfjell, \mu = ' + mu_kf);

```

2c

```
% Check graphs for answer
% Dist. have no overlap and Kvitfjell is more probable to have more
  snow!
```

3 MMSE and MAP estimator for Gaussian mixture posteriors

```
close all; clc;

% Weights
w_a = [0.1 0.9];
w_b = [0.49 0.51];
w_c = [0.4 0.6];

% Means
mu_a = [1 1];
mu_b = [5 -5];
mu_c = [1 2];

% Standard deviations
sigma_a = [sqrt(0.5) sqrt(9)];
sigma_b = [sqrt(2) sqrt(2)];
sigma_c = [sqrt(2) sqrt(1)];

% Calculate the posterior by adding the individual dist.
x = -10:0.01:10;
pa = w_a(1)*normpdf(x, mu_a(1), sigma_a(1)) + w_a(2)*normpdf(x,
    mu_a(2), sigma_a(2));
pb = w_b(1)*normpdf(x, mu_b(1), sigma_b(1)) + w_b(2)*normpdf(x,
    mu_b(2), sigma_b(2));
pc = w_c(1)*normpdf(x, mu_c(1), sigma_c(1)) + w_c(2)*normpdf(x,
    mu_c(2), sigma_c(2));

% Calculate the MMSEE
xHat_a = gaussMixMMSEEst(w_a, mu_a, sigma_a);
xHat_b = gaussMixMMSEEst(w_b, mu_b, sigma_b);
xHat_c = gaussMixMMSEEst(w_c, mu_c, sigma_c);

% Plot the posterior distributions
d = plot(x, [pa; pb; pc]);
hold on
set(d(1), 'Color', 'r');
set(d(2), 'Color', 'g');
set(d(3), 'Color', 'b');

% Plot MMSE
xline(xHat_a, '--r');
```

```
xline(xHat_b, '--g');  
xline(xHat_c, '--b');  
  
% Plot MAP  
plot(x(pa == max(pa)), 0, 'or')  
plot(x(pb == max(pb)), 0, 'og')  
plot(x(pc == max(pc)), 0, 'ob')  
  
legend('a', 'b', 'c', 'a_{MMSE}', 'b_{MMSE}', ...  
      'c_{MMSE}', 'a_{MAP}', 'b_{MAP}', 'c_{MAP}');
```

Published with MATLAB® R2019b