

Problem 1 (graded by Yiran) 50 points**(a) - 8 points**

With the misfit function defined as

$$F(\mathbf{m}) = \sum_i \frac{(d_i - g_i(\mathbf{m}))^2}{2\sigma_d^2}$$

The posterior PDF

$$\begin{aligned} P(\mathbf{m}|\mathbf{d}) \propto P(\mathbf{d}|\mathbf{m}) &\propto \exp(-F(\mathbf{m})) \\ &\approx \exp\left(-F(\mathbf{m}_0) - \nabla F(\mathbf{m}_0)^T(\mathbf{m} - \mathbf{m}_0) - \frac{1}{2}(\mathbf{m} - \mathbf{m}_0)^T \mathbf{H}(\mathbf{m}_0)(\mathbf{m} - \mathbf{m}_0)\right) \end{aligned}$$

has the form of a multivariate Gaussian distribution, and the covariance matrix is

$$\mathbf{V} = \text{cov}(\mathbf{m}) = \mathbf{H}(\mathbf{m}_0)^{-1}$$

Note we want to know the covariance matrix when \mathbf{m}_0 is the least squares solution.

In HW2, we didn't include the data error, and

$$\hat{G}_{i,k} = \frac{\partial g_i}{\partial m_k}$$

To include the data error

$$\mathbf{H} \approx \frac{\hat{\mathbf{G}}^T \hat{\mathbf{G}}}{\sigma_d^2}$$

So we need to modify the Hessian in HW2 by dividing by σ_d^2 when we calculate the model covariance matrix.

(See MATLAB code) The output model covariance matrix is:

0.0101	-0.0123	0.0095	0.0548
-0.0123	0.0198	-0.0156	-0.0853
0.0095	-0.0156	0.0237	0.1034
0.0548	-0.0853	0.1034	0.4985

(b)- 8 points

The covariance matrix calculated here will be similar to the Monte Carlo simulation in HW2.

Diagonal elements show the variance of each parameter. The square root of the diagonal elements gives the standard deviations: $\sigma_{x_s} = 0.1006$, $\sigma_{y_s} = 0.1405$, $\sigma_{z_s} = 0.1538$, $\sigma_P = 0.7060$. The number is very close to that estimated before: $\sigma_{x_s} = 0.099670$, $\sigma_{y_s} = 0.137469$, $\sigma_{z_s} = 0.149719$, $\sigma_P = 0.691071$.

The off diagonal shows the covariance between parameters. We can scale the covariance matrix to correlation matrix ($\rho_{xy} = \frac{\sigma_{xy}}{\sigma_x \sigma_y}$). The correlation matrix is:

1.0000	-0.8685	0.6127	0.7721
-0.8685	1.0000	-0.7200	-0.8598
0.6127	-0.7200	1.0000	0.9520
0.7721	-0.8598	0.9520	1.0000

we see that there are strong trade-offs between model parameters. The strong negative correlation between x_s and y_s , for example, is also shown in the plot in HW2.

(c)- 8 points

Now

$$F(\mathbf{m}) = \frac{1}{2\sigma^2}(\mathbf{d} - \mathbf{G}\mathbf{m})^T(\mathbf{d} - \mathbf{G}\mathbf{m})$$

where σ is the data error, and

$$\begin{aligned}\mathbf{G} &= \begin{bmatrix} \mathbf{1} & \mathbf{x} \end{bmatrix} \\ \mathbf{m} &= [m_1, m_2]^T\end{aligned}$$

The Hessian is

$$\mathbf{H} = \frac{1}{\sigma^2}\mathbf{G}^T\mathbf{G}$$

and

$$\text{cov}(\mathbf{m}) = \mathbf{H}^{-1}$$

In this problem, the data error is strongly underestimated. Your friend tells you his experimental error on y are all ≈ 0.1 , it's possible that he only tells the instrument error. There are other sources of random errors, for example, the environmental factors, which can be large.

We need to re-estimate the data error, and calculate the model covariance matrix based on the new data error.

Let s be our estimate of the data error, $n = 30$ the number of observations, $m = 2$ the number of model parameters, \mathbf{m}_{L_2} the least squares solution.

Then, from equation (2.63) in Aster

$$s = \sqrt{\frac{1}{n-m} \|\mathbf{d} - \mathbf{G}\mathbf{m}_{L_2}\|_2^2} \approx 26.3$$

(see MATLAB code) The output covariance matrix is:

$$\begin{bmatrix} 23.7090 & -0.1283 \\ -0.1283 & 0.0249 \end{bmatrix}$$

(*) Result using given data error $\sigma = 0.1$

(Not a correct answer, just for your reference)

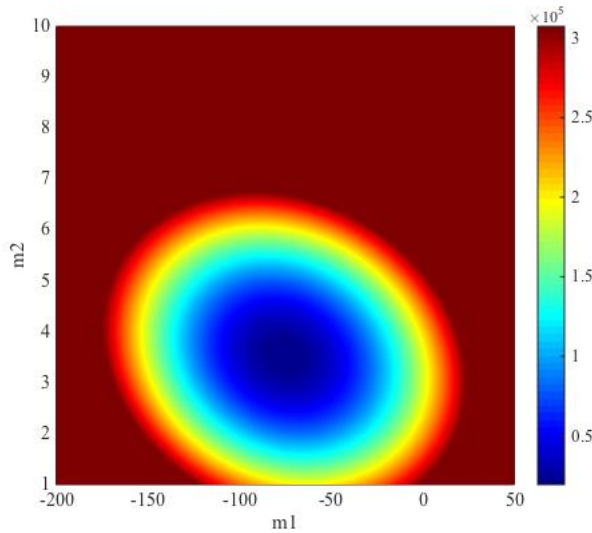
(see MATLAB code) The output covariance matrix is:

$$\begin{bmatrix} 1.0\text{e-}03 & * \\ 0.3429 & -0.0019 \\ -0.0019 & 0.0004 \end{bmatrix}$$

(d)- 8 points

The model covariance is calculated from the distribution $P(\mathbf{m}|\mathbf{d})$. The square root of the eigenvalues and eigenvectors of the model covariance matrix measure the shape (length and direction of the semiaxes) of the isocontour of $P(\mathbf{m}|\mathbf{d})$. The shape of the isocontour of $P(\mathbf{m}|\mathbf{d}) \propto P(\mathbf{d}|\mathbf{m}) \propto \exp(-F(\mathbf{m}))$ is scaled to the isocontour of the error ellipse $F(\mathbf{m})$, which we plotted in HW2.

The eigenvectors of the covariance matrix are: $\begin{bmatrix} -0.0054 & -1.0000 \end{bmatrix}$ and $\begin{bmatrix} -1.0000 & 0.0054 \end{bmatrix}$, correspond to an angle of 89.7° and 179.7° measured clockwise from x axis - almost a horizontal ellipse; the ratio of the length of the semiaxis is $\sqrt{\lambda_1}/\sqrt{\lambda_2} = 0.0319$. We see that it is true for the error ellipse as shown below (notice the difference in the scale of x and y axis). The negative correlation between m_1 and m_2 is also clear from the off-diagonal element of the model covariance matrix, the analysis on the eigenvectors, and the error ellipse.

**(e)- 10 points**

First calculate the misfit $F(\mathbf{m})$ for the full parameter space.

From Bayesian law and uniform prior,

$$\begin{aligned} P(\mathbf{m}|\mathbf{d}) &\propto P(\mathbf{d}|\mathbf{m}) \\ &\propto \exp(-F(\mathbf{m})) \end{aligned}$$

Normalize the pdf so its integral over the model space equals 1.

$$P(\mathbf{m}|\mathbf{d}) = \frac{\exp(-F(\mathbf{m}))}{\int \exp(-F(\mathbf{m})) d\mathbf{m}}$$

that is

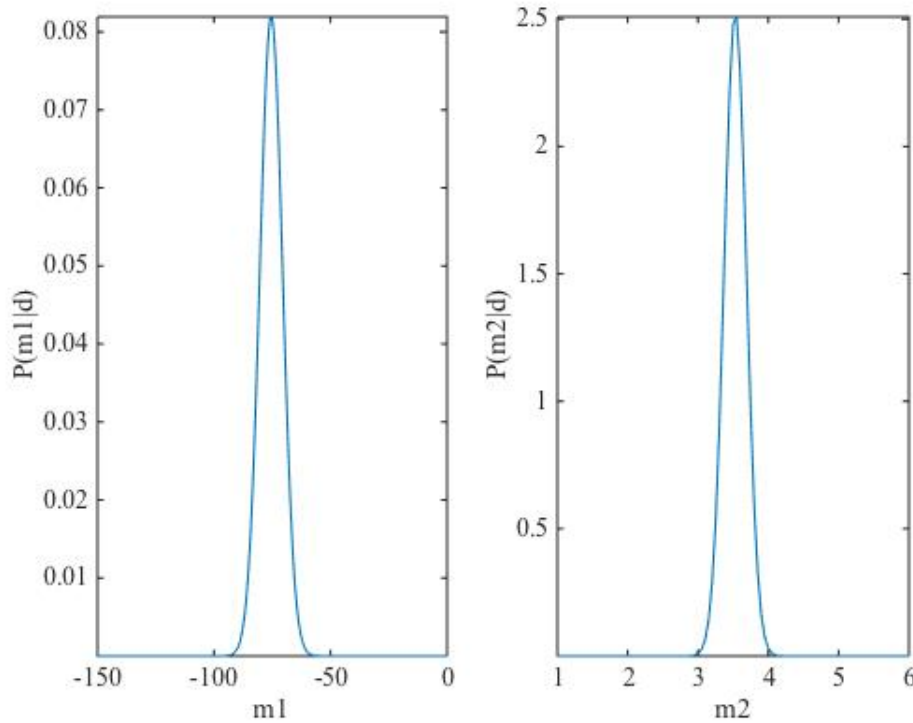
$$P(m_1, m_2|\mathbf{d}) = \frac{\exp(-F(m_1, m_2))}{\int \int \exp(-F(m_1, m_2)) dm_1 dm_2}$$

The marginals are:

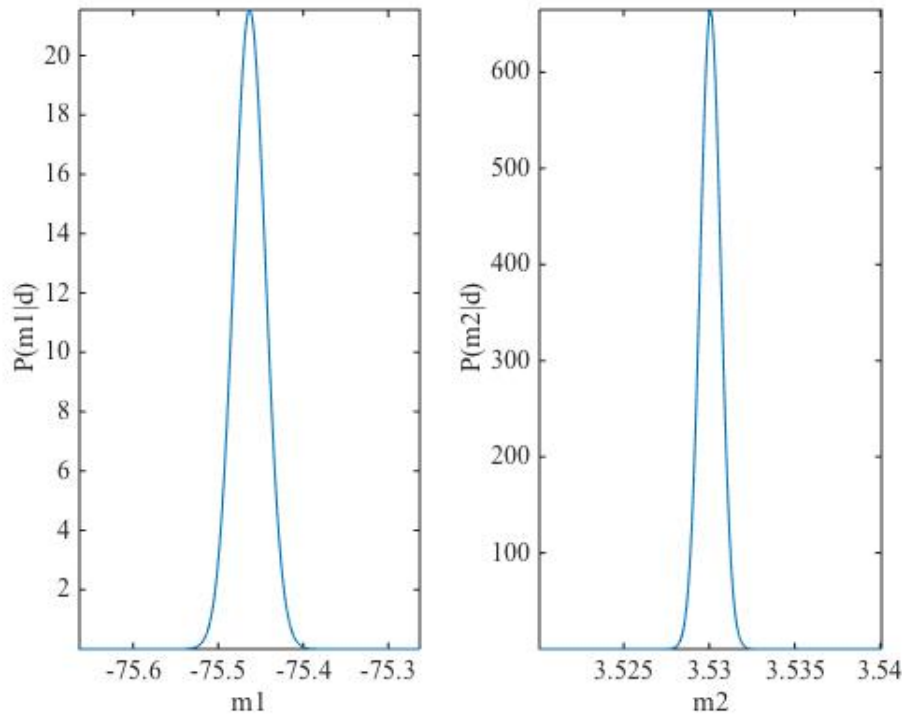
$$P(m_1|d) = \int P(m_1, m_2|d) dm_2$$

$$P(m_2|d) = \int P(m_1, m_2|d) dm_1$$

We can approximate the integral with the midpoint numerical integration: $\int f(x)dx = \sum_i f(x_i)\Delta x$.
Using estimated data error $s = 26.3$, we have



(*) Result using given data error $\sigma = 0.1$
(Not a correct answer, just for your reference)



(f)- 8 points

By definition

$$\sigma_{m_i}^2 = E[m_i^2] - (E[m_i])^2 = \int p(m_i|d)m_i^2 dm_i - \left(\int p(m_i|d)m_i dm_i \right)^2$$

(see MATLAB code)

Using estimated data error $s = 26.3$, we get $\sigma_{m_1} = 4.8692$, $\sigma_{m_2} = 0.1577$, which are very close to square root of the diagonal elements of the covariance matrix calculated in (c).

(*) Result using given data error $\sigma = 0.1$

(Not a correct answer, just for your reference)

We get $\sigma_{m_1} = 0.0185$, $\sigma_{m_2} = 0.0006$, which are very close to square root of the diagonal elements of the covariance matrix calculated in (c).

MATLAB code for (a)-(b)

```

1 function hw2_1
2 sigma = 0.001;
3 % std_mc = [0.099670 0.137469 0.149719 0.691071]';
4 %%hw 2 problem 1d
5 xi = [0 11 15 6 -7 3]';
6 yi = [0 0 6 13 10 -7]';
7 ui = [0.103 0.162 0.065 0.036 0.025 0.169]';
8 M0 = [8 -5 10 30]'; %initial guess, xs ys zs P
9
10 [Ms, mcov] = nonlinear_solver(xi, yi, ui, M0, sigma);
11
12 % model covariance matrix
13 disp('model covariance matrix:');
14 disp(mcov);
15 disp('standard deviation');
16 std_m = sqrt(diag(mcov));
17
18 % calculate correlation matrix

```

```

19 s = diag(std_m);
20 disp('coefficient matrix:');
21 mcoef = s^(-1)*mcov*s^(-1)
22 end
23
24 function [M, mcov]=nonlinear_solver(x,y,ui,Minit,sigma)
25
26 M = Minit;
27
28 misfit_old = 0;
29
30 misfit = 0;
31
32 for ii = 1:1:1000
33
34     misfit = compute_misfit(x,y,M,ui);
35
36
37     [Grad,Hess] = compute_gradient_approx_hess(x,y,M,misfit);
38
39     deltaM= (Hess)\Grad';
40
41     M=M + deltaM;
42
43     if (norm(misfit - misfit_old)<1e-7)
44         break;
45     end
46
47     misfit_old = misfit;
48 end
49
50 mcov = inv(Hess/sigma^2);
51
52 %disp(['Number of iterations:',num2str(ii)]);
53 end
54
55 function [ misfit ] = compute_misfit( x,y,M,ui )
56
57 xs = M(1);
58 ys = M(2);
59 zs = M(3);
60 p = M(4);
61
62 misfit = ui - p*zs./((x - xs).^2 + (y - ys).^2 + zs^2).^(3/2);
63
64 end
65
66 function [ Grad, Hess] = compute_gradient_approx_hess( x,y,M,misfit )
67
68 xs = M(1);
69 ys = M(2);
70 zs = M(3);
71 p = M(4);
72
73 eta = ((x - xs).^2 + (y - ys).^2 + zs^2);
74
75 dx = x-xs;
76 dy = y-ys;
77
78 Ghat(:,1) = (3.*p.*zs.*(dx))./((eta).^(5/2));
79 Ghat(:,2) = (3.*p.*zs.*(dy))./((eta).^(5/2));
80 Ghat(:,3) = p./(eta).^(3/2) - (3*p.*zs.^2)./(eta).^(5/2);
81 Ghat(:,4) = zs./(eta).^(3/2);
82
83 Grad = (misfit')* Ghat;
84
85 Hess = (Ghat')*Ghat;
86 %this is the approximated Hessian;
87 end

```

MATLAB code for (c)-(f)

```

1 function hw2_2s
2 % load data
3 load gel18_hw2.mat
4 sigma = 0.1; % data error
5 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
6 % least squares
7 disp('*** start least squares ***');
8 for i = 1:2
9     if i == 1
10         disp('(1) use given data error = 0.1');
11         data_err = sigma;
12     elseif i == 2
13         disp('(2) use estimated data error');
14         % estimate the data error
15         s = sqrt( ( misfit(x,y,m1_ls,m2_ls,2)/(length(x)-2) ) );
16         disp('estimate data error');
17         disp(s);
18         data_err = s;
19     end
20
21     [m1_ls,m2_ls,mcov] = least_square(x,y,data_err);
22
23     disp('model covariance matrix:');

```

```

24     disp(mcov);
25     % [V,D] = eig(mcov)
26     disp('standard deviation of m1 and m2:');
27     disp(sqrt(diag(mcov)));
28 end
29 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
30 % grid search
31 disp('*** start monte carlo ***');
32
33 for i = 1:2
34     if i == 1
35         disp('(1) use given data error = 0.1');
36         % narrow the searching range
37         % so dm can be small enough to show the marginal pdf
38         m1 = (m1_ls - 0.2):0.0005:(m1_ls + 0.2);
39         m2 = (m2_ls - 0.01):0.0001:(m2_ls + 0.01);
40         data_err = sigma;
41     elseif i == 2
42         disp('(2) use estimated data error');
43         data_err = s;
44         m1 = -200:0.1:50;
45         m2 = 1:0.05:6;
46     end
47     dm1 = m1(2) - m1(1);
48     dm2 = m2(2) - m2(1);
49
50     % L2 norm
51     [m1_l2,m2_l2,err_l2,err_all_l2] = grid_search(x,y,m1,m2,2);
52
53     % joint pdf
54     p_mlm2 = exp(-(err_all_l2-min(err_all_l2(:)))/(2*data_err^2));
55     p_mlm2 = p_mlm2/(sum(p_mlm2(:))*dm1*dm2);
56
57     % marginal
58     p_m1 = dm2 * sum(p_mlm2,2);
59     p_m2 = dm1 * sum(p_mlm2,1);
60
61     m1 = m1(:);
62     m2 = m2(:);
63     p_m1 = p_m1(:);
64     p_m2 = p_m2(:);
65
66     figure;
67     subplot(121);
68     plot(m1,p_m1);
69     xlabel('m1'); ylabel('P(m1|d)'); axis tight;
70     subplot(122);
71     plot(m2,p_m2);
72     xlabel('m2'); ylabel('P(m2|d)'); axis tight;
73
74     % \sigma m1 m2
75     sigma_m1 = sqrt(dm1 * sum(p_m1 .* m1.^2) - (dm1 * sum(p_m1 .* m1)).^2);
76     sigma_m2 = sqrt(dm2 * sum(p_m2 .* m2.^2) - (dm2 * sum(p_m2 .* m2)).^2);
77     disp('standard deviation of m1 and m2:');
78     disp([sigma_m1 sigma_m2]);
79 end
80 end
81
82 % LEAST SQUARE SOLUTION
83 function [m1, m2, mcov] = least_square(xdata,ydata,sigma)
84 G = [ones(length(xdata),1) xdata(:)];
85 tmp = (G'*G)^(-1) * G' * ydata;
86 m1 = tmp(1);
87 m2 = tmp(2);
88 mcov = inv(G'*G/sigma^2);
89 end
90
91 % GRID SEARCH
92 function [m1_best,m2_best,err_best,err] = grid_search(xdata,ydata,m1,m2,flag)
93 % error over the model space
94 err = zeros(length(m1),length(m2));
95 for i = 1:length(m1)
96     for j = 1:length(m2)
97         err(i,j) = misfit(xdata,ydata,m1(i),m2(j),flag);
98     end
99 end
100
101 % find the optimum solution
102 [err_best,id] = min(err(:));
103 [I,J] = ind2sub(size(err),id);
104 m1_best = m1(I);
105 m2_best = m2(J);
106 end
107
108 % CALCULATE THE MISFIT
109 function err = misfit(xdata,ydata,m1,m2,flag)
110 ypred = m1 + m2 * xdata;
111 if flag == 1 % L1 norm
112     err = sum(abs(ypred-ydata));
113 elseif flag == 2 % L2 norm
114     err = sum((ypred-ydata).^2);
115 end
116 end

```

Problem 2 (graded by Kangchen) - 50 points**(a) - 5 points**

It is not reasonable to assume that all model parameters have constant priors $(-\infty \text{ to } \infty)$. For x_s and y_s , we should know that it cannot be too far from where the volcano is. For z_s which we defined as depth, it must be positive. For P which is related to pressure, we know that it must be positive too.

(b) - 10 points

We can incorporate this information as a prior by multiplying our likelihood $P(\{d_k\}|\mathbf{m})$ by a prior for the model $P(\mathbf{m})$. In this case we will use a Gaussian distribution for model parameter p with $\mu = 35$ and $\sigma = 6$.

$$\begin{aligned} P(\mathbf{m}) &= e^{-\frac{1}{2}(\frac{p-\mu}{\sigma})^2} \\ P(\mathbf{m}|\{d_k\}) &= P(\{d_k\}|\mathbf{m})P(\mathbf{m}) \\ P(\mathbf{m}|\{d_k\}) &= e^{-F(\mathbf{m})}e^{-\frac{1}{2}(\frac{p-\mu}{\sigma})^2} \\ P(\mathbf{m}|\{d_k\}) &= e^{-F(\mathbf{m})-\frac{1}{2}(\frac{p-\mu}{\sigma})^2} \end{aligned}$$

(c.i) - 5 points

$$\begin{aligned} \int_{\phi_1}^{\phi_2} P(\phi)d\phi &= \int_{\phi_1}^{\phi_2} \frac{1}{\phi}d\phi = \ln(\phi_2) - \ln(\phi_1) \\ \int_{k\phi_1}^{k\phi_2} P(\phi)d\phi &= \int_{k\phi_1}^{k\phi_2} \frac{1}{\phi}d\phi = \ln(k\phi_2) - \ln(k\phi_1) \\ &= \ln(k) + \ln(\phi_2) - \ln(k) - \ln(\phi_1) \\ &= \ln(\phi_2) - \ln(\phi_1) \end{aligned}$$

So $P(\phi) = \frac{1}{\phi}$ satisfies the scale independent criterion.

(c.ii) - 5 points

We can again incorporate this information as a prior in our expression for $P(\mathbf{m}|\{d_k\})$ by multiplying it times our likelihood (just like in part b). Our expression becomes:

$$P(\mathbf{m}|\{d_k\}) = e^{-F(\mathbf{m})}\frac{1}{p} = e^{-F(\mathbf{m})-\ln(p)}$$

(d) - 5 points

We want to incorporate both independent pieces of information into our prior. In this case we can multiply the two priors together. Our new expression becomes:

$$P(\mathbf{m}|\{d_k\}) = \frac{1}{p}e^{-F(\mathbf{m})}e^{-\frac{1}{2}(\frac{p-\mu}{\sigma})^2} = e^{-F_{old}(\mathbf{m})-\frac{1}{2}(\frac{p-\mu}{\sigma})^2-\ln(p)}$$

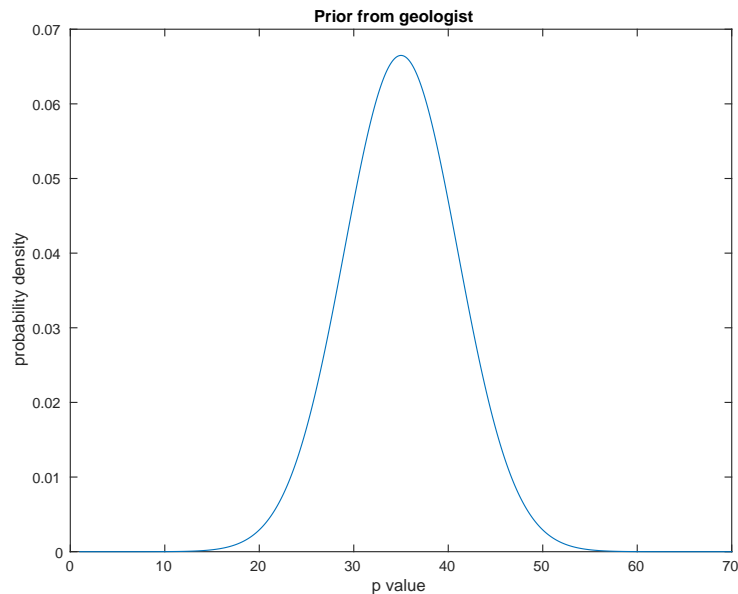


Figure 0.1: Plot of the Gaussian prior for p .

Let's plot the two priors separately and then together to see which information dominates the result. Figure 0.1 is the Gaussian prior. Figure 0.2 is the scale independent prior. Figure 0.3 is their combination. We can see that the Gaussian prior is scaled by the scale invariant prior, but still mainly retains its shape and thus dominates the result.

(e) - 5 points

It doesn't matter in what order we do things as long as priors for analysis are not biased by our data (and they shouldn't be) and your experiment is not affected by prior information. An example of a situation in which the experiment is affected by prior is that if we have some prior information about x_s and y_s which is not the case here, we can conduct our measurements closer to those prior points which may increase accuracy of our inversion.

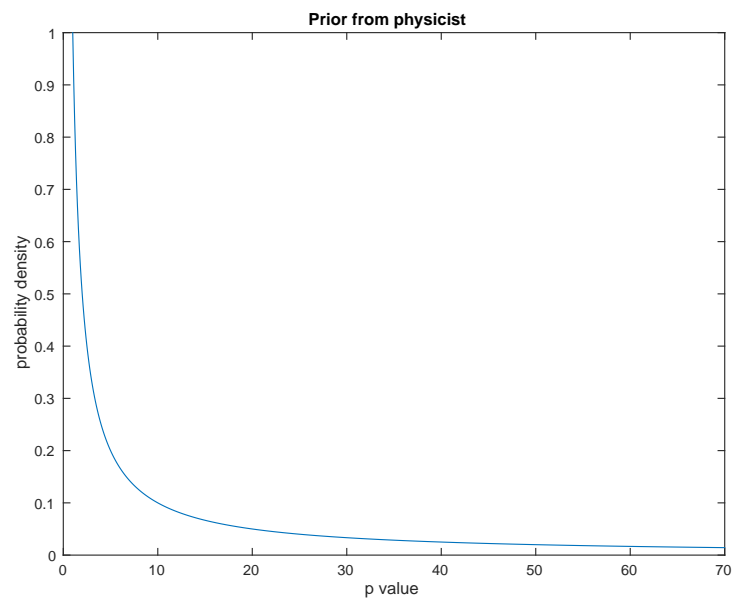


Figure 0.2: Plot of the scale invariant prior for p .

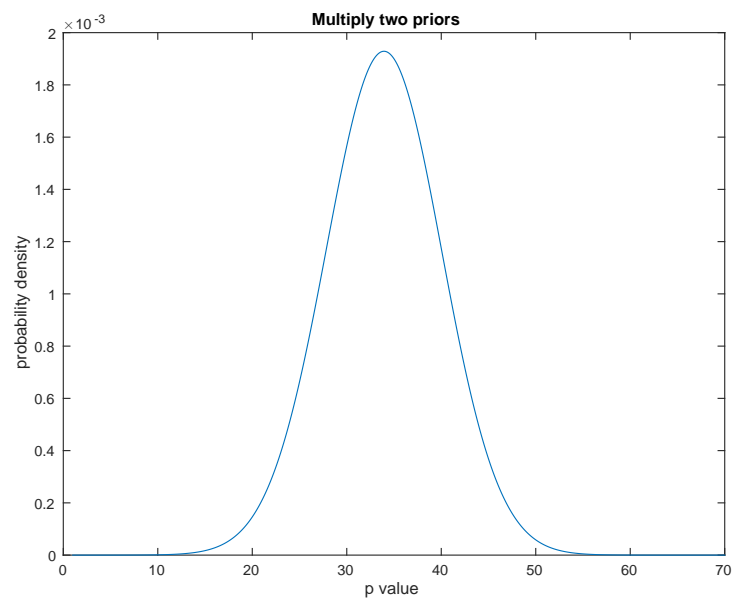


Figure 0.3: Plot of combination of the Gaussian and scale invariant priors for p .

(f) - 15 points

Previous expression: $P_{old}(\mathbf{m}|\{d_k\}) = e^{-F_{old}(\mathbf{m})}$

New expression: $P(\mathbf{m}|\{d_k\}) = \frac{1}{p} e^{-F_{old}(\mathbf{m})} e^{-\frac{1}{2}(\frac{p-\mu}{\sigma})^2}$

Previous misfit function: $F_{old}(\mathbf{m})$

To find the new misfit function we will need to manipulate our expression to get everything into a single exponential. Let's start with the $\frac{1}{p}$ part:

$$\frac{1}{p} = e^{\ln(\frac{1}{p})} = e^{-\ln(p)}$$

$$P(\mathbf{m}|\{d_k\}) = e^{-\ln(p)} e^{-F_{old}(\mathbf{m})} e^{-\frac{1}{2}(\frac{p-\mu}{\sigma})^2}$$

New misfit function: $F(\mathbf{m}) = F_{old}(\mathbf{m}) + \ln(p) + \frac{1}{2}(\frac{p-35}{6})^2$

We only need to make a couple simple changes to our code from HW2. Instead of just using the L2 norm as our error function, we now need to add our extra two terms to our misfit function:

We will add a term to the last entries of γ and the Hessian to account for the new priors on p .

Add the first derivative of our new part of the misfit to γ :

$$\gamma = \gamma_{old} - \frac{1}{p} - \frac{p - \mu}{\sigma^2}$$

And the second derivative to the last entry of the Hessian:

$$\frac{\partial^2 F}{\partial p^2} = \frac{\partial^2 F_{old}}{\partial p^2} + \frac{-1}{p^2} + \frac{1}{\sigma^2}$$

Here is the full code:

```

1 %%problem 2f
2 xi = [0 11 15 6 -7 3]';
3 yi = [0 0 6 13 10 -7]';
4 ui = [0.103 0.162 0.065 0.036 0.025 0.169]';
5 M0 = [8 -5 10 30]'; %initial guess
6 Ms = nonlinear_solver(xi,yi,ui,M0);

1 function [ Gamma, Hess] = compute_gradient_approx_hess( x,y,M, misfit )
2
3 xs = M(1);
4 ys = M(2);
5 zs = M(3);
6 p = M(4);
7
8 eta = ((x - xs).^2 + (y - ys).^2 + zs^2);
9
10 dx = x-xs;
11 dy = y-ys;
12
13 Ghat(:,1) = (3.*p.*zs.*(dx))./((eta).^(5/2));
14 Ghat(:,2) = (3.*p.*zs.*(dy))./((eta).^(5/2));
15 Ghat(:,3) = p./((eta).^(3/2)) - (3*p.*zs.^2)./(eta).^(5/2);
16 Ghat(:,4) = zs./((eta).^(3/2));
17
18 sigma_P = 6;
19
20 P_mean = 35;
21
22 Gamma = -(misfit') * Ghat;
```

```

23 Gamma = Gamma + [0 0 0 1/p + (p-P_mean)/(sigma_P^2)];
24
25 Hess = (Ghat')*Ghat ;
26
27 Hess(4,4) = Hess(4,4) - 1/p^2 + 1/(sigma_P^2) ;
28
29 %this is the approximated Hessian;
30
31 end
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

Our best fit solution is now:

$$\begin{bmatrix} x_s \\ y_s \\ z_s \\ p \end{bmatrix} = \begin{bmatrix} 8.5332 \\ -5.7685 \\ 12.1910 \\ 33.9391 \end{bmatrix}$$

Comparing with old solution, the new solution has p value closer to 35 which is given by geologist's prior.

Problem 3 (graded by Kangchen) - bonus 15 points**(a)-5 points**

The posterior probability is given by

$$p(\mu|x) \propto \left(\sum_{k=1}^N (x_k - \mu)^2 \right)^{-\frac{N-1}{2}} = \exp \left(-\frac{N-1}{2} \ln \left[\sum_{k=1}^N (x_k - \mu)^2 \right] \right). \quad (0.1)$$

Minima of F maximize the probability density function. We have that

$$F(\mu) = \frac{N-1}{2} \ln \left(\sum_{k=1}^N (x_k - \mu)^2 \right), \quad (0.2)$$

$$\frac{\partial F}{\partial \mu} = -(N-1) \frac{\sum_{k=1}^N (x_k - \mu)}{\sum_{k=1}^N (x_k - \mu)^2} \quad (0.3)$$

$$\frac{\partial^2 F}{\partial \mu^2} = (N-1) \frac{N \sum_{k=1}^N (x_k - \mu)^2 - 2 \left(\sum_{k=1}^N (x_k - \mu) \right)^2}{\left(\sum_{k=1}^N (x_k - \mu)^2 \right)^2}. \quad (0.4)$$

Taking the derivative with respect to μ and setting it to zero yields the minimum μ_0

$$\mu_0 = \frac{1}{N} \sum_{k=1}^N x_k, \quad (0.5)$$

For the second derivate at the best fit solution we have

$$\frac{\partial^2 F}{\partial \mu^2}(\mu = \mu_0) = \frac{N(N-1)}{\sum_{k=1}^N (x_k - \mu_0)^2}. \quad (0.6)$$

The standard deviation σ_μ then amounts to

$$\sigma_\mu = \frac{1}{F''(\mu_0)^{1/2}} = \frac{S}{\sqrt{N}}, \quad (0.7)$$

where

$$S = \sqrt{\frac{1}{N-1} \sum_{k=1}^N (x_k - \mu_0)^2} \quad (0.8)$$

is the sample variance of the data.

(b)-5 points

We assume $\mathbf{Q} = [\mathbf{e}_1, \mathbf{e}_2]$ where $\|\mathbf{e}_1\| = 1, \|\mathbf{e}_2\| = 1$. We can write $\mathbf{e}_1 = (\cos(\theta), \sin(\theta))^T$ and $\mathbf{e}_2 = (\cos(\phi), \sin(\phi))^T$. Since \mathbf{Q} is an orthogonormal matrix, \mathbf{e}_1 and \mathbf{e}_2 are orthogonal. That means $\cos(\theta)\cos(\phi) + \sin(\theta)\sin(\phi) = \cos(\theta - \phi) = 0$. So $\theta - \phi = \pi/2$ or $-\pi/2$. If we assume that $\phi = \theta + \pi/2$, then $\mathbf{Q} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}$. This can be understood as rotating a vector by θ degree

counterclockwise. Another way to interpret what \mathbf{Q} did is that $\mathbf{Q}^T \begin{pmatrix} x \\ y \end{pmatrix}$ gives the projection of

$\begin{pmatrix} x \\ y \end{pmatrix}$ onto \mathbf{e}_1 and \mathbf{e}_2 where $\mathbf{Q} = (\mathbf{e}_1, \mathbf{e}_2)$.

(c)-5 points

We need to show that $\sigma_x^2 = \frac{B}{AB-C^2}$. We are dealing with an integral of the f

$$\sigma_x^2 = \frac{\int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} dx dy x^2 \exp\left(-\frac{1}{2}x^T A x\right)}{\int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} dx dy \exp\left(-\frac{1}{2}x^T A x\right)} \quad (0.9)$$

$$= \frac{\int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} dx dy x^2 \exp\left(-\frac{1}{2}(Ax^2 + 2Cxy + By^2)\right)}{\int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} dx dy \exp\left(-\frac{1}{2}(Ax^2 + 2Cxy + By^2)\right)} \quad (0.10)$$

$$= \frac{\int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} dx dy x^2 \exp\left(-\frac{1}{2}(A - C^2/B)x^2 - \frac{1}{2}B(y + Cx/B)^2\right)}{\int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} dx dy \exp\left(-\frac{1}{2}(A - C^2/B)x^2 - \frac{1}{2}B(y + Cx/B)^2\right)} \quad (0.11)$$

$$= \frac{\int_{-\infty}^{+\infty} dx x^2 \exp\left(-\frac{1}{2}(A - C^2/B)x^2\right) \int_{-\infty}^{+\infty} dy \exp\left(-\frac{1}{2}B(y + Cx/B)^2\right)}{\int_{-\infty}^{+\infty} dx \exp\left(-\frac{1}{2}(A - C^2/B)x^2\right) \int_{-\infty}^{+\infty} dy \exp\left(-\frac{1}{2}B(y + Cx/B)^2\right)} \quad (0.12)$$

$$= \frac{\int_{-\infty}^{+\infty} dx x^2 \exp\left(-\frac{1}{2}(A - C^2/B)x^2\right)}{\int_{-\infty}^{+\infty} dx \exp\left(-\frac{1}{2}(A - C^2/B)x^2\right)} \quad (0.13)$$

Note that from 0.12 to 0.13:

$$\int_{-\infty}^{+\infty} dy \exp\left(-\frac{1}{2}B(y + Cx/B)^2\right) = \int_{-\infty+Cx/B}^{+\infty+Cx/B} dy' \exp\left(-\frac{1}{2}By'^2\right) = \int_{-\infty}^{+\infty} dy' \exp\left(-\frac{1}{2}By'^2\right)$$

So that integral is not related to x . That's why we can cancel out that term from both numerator and denominator.

If we set $\sigma^2 = 1/(A - C^2/B)$, we arrive at

$$\sigma_x^2 = \frac{\int dx x^2 \exp\left(-\frac{1}{2}x^2/\sigma^2\right)}{\int dx \exp\left(-\frac{1}{2}x^2/\sigma^2\right)} \quad (0.14)$$

, which as we have seen in previous homework sets and solutions implies

$$\sigma_x^2 = \langle x^2 \rangle = \sigma^2 = \frac{1}{A - \frac{C^2}{B}} = \frac{B}{AB - C^2}. \quad (0.15)$$