3. a) what happened to figure 3????

3.

b) i.

Requires we specify inputs, X, then we have

p(f(X)) = N(f(X); 0, k(X, X))

So we calculate k(X, X), and this is enough to specify the MVN fully. Take a sample from this MVN (there’s a numpy function for this so I ill assume that’s doable). This gives us a sample of f(X). We can now plot X against f(X)

b) ii.

σf → 0 : not sure?

l → 0 : influence of a single data point only spreads over very small distance. ‘Spikes’ at each data point, otherwise 0.

Optimise the marginal likelihood.

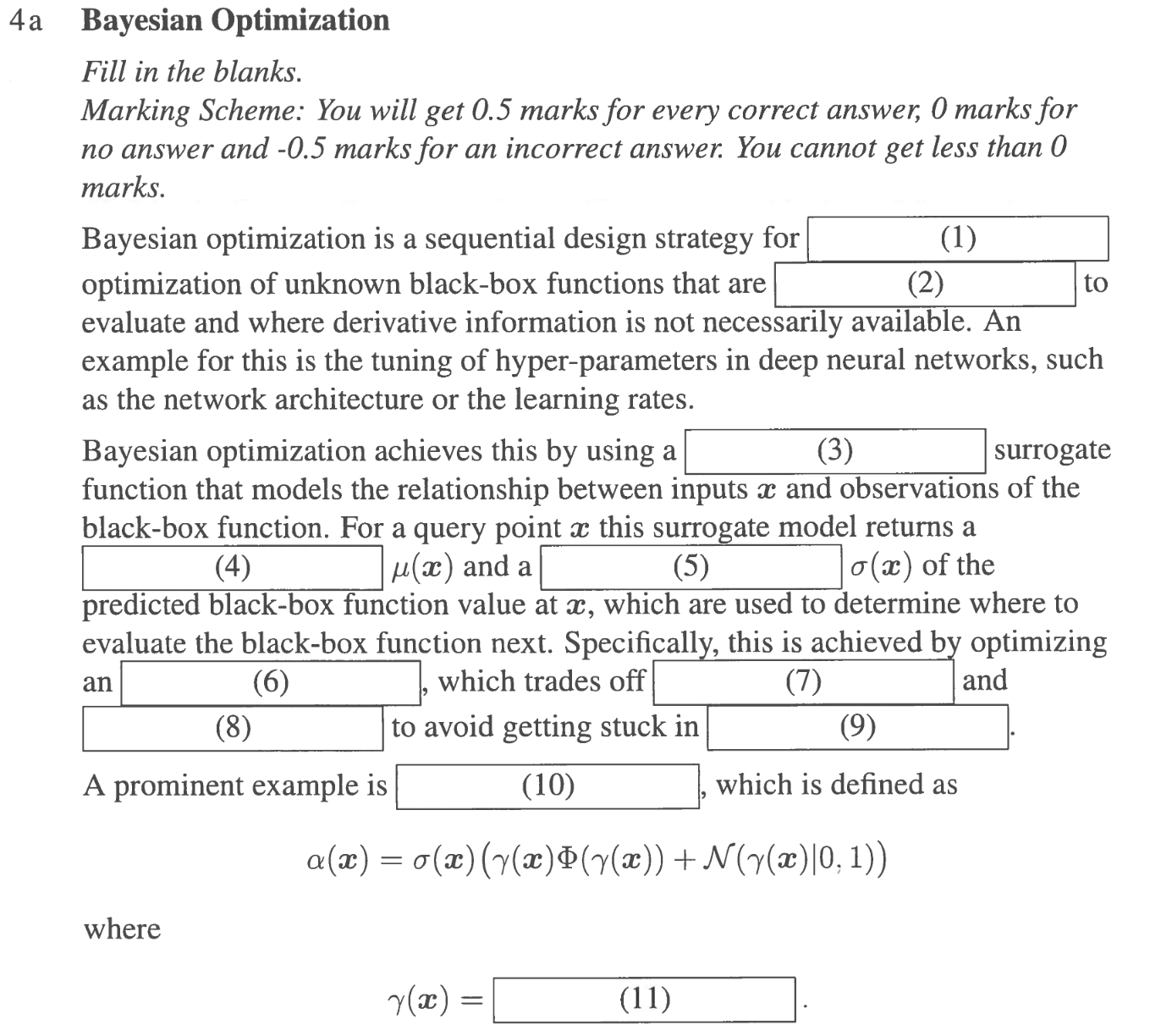
b) iii.

Not sure what they’re going for here but:

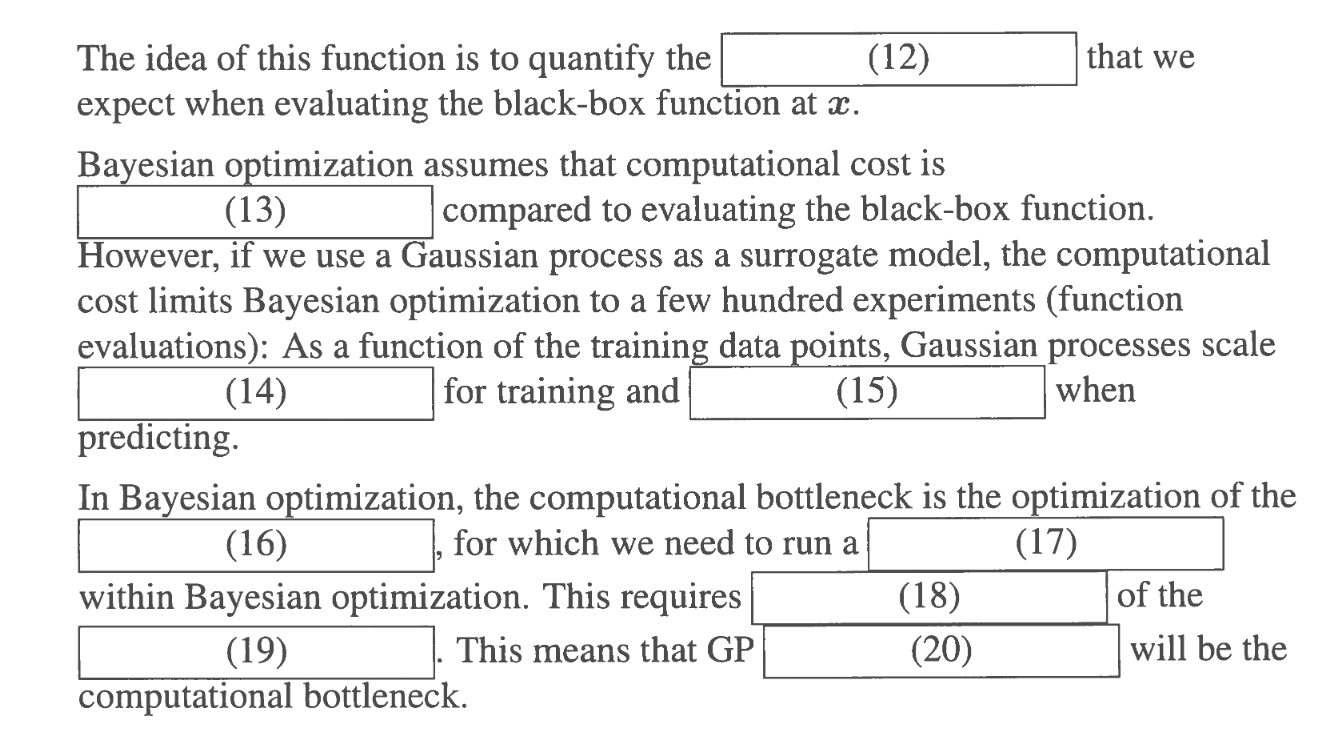
use kernel: k1 + k2

This is more flexible as it can allow for either kernel (by tuning hyperparameters - one or the other can go to 0), or both, which can account for different data patterns.

4a



(1) global (or 0-th order)  
(2) expensive  
(3) ~~modeled~~ ~~(?)~~ probabilistic  
(4) mean  
(5) variance  
(6) acquisition function  
(7) exploration  
(8) exploitation  
(9) local minima (?)  
(10) expected improvement   
(11) f(x\_best) - mean(x) / var(x)^0.5



(12) improvement in the best value of f   
(13) smaller (I assume here it is talking about the model)  
(14) O(n^3) (?)  
(15) O(n^2) (?)  
(16) acquisition function  
(17) global optimization / dense sampling  
(18) evaluation  
(19) model  
(20) prediction