# Machine Learning and optimization Project

Taniya

February 18, 2021

## 1 Question 1

In the first part of the problem our main motive is to compare different approaches of modelling such as different kernels, different GP configurations(different lengthscale and different variance). For the comparison we have used different train sets and test sets to evaluate the performances of each modeling approach. The measure of performances can be done by the use of RMSE, R2 score, MNLL, Std RMSE and as we know the best combinations from all is that which have the r2 value very near to 1 . In our sampling we have used 7 different kernels with lengthscales are 1, 0.1 ,10. And we also ovbserved that with the increase in the number of points errors such as RMSE, MNLL and Std RMSE decreases for most of combinations but this is also increase for very few cases in which kernels not nicely fit to our problem.

| kernel name | pts | ls | var | R2 score | RMSE | MNLL | Std RMSE |
|---|---|---|---|---|---|---|---|
| RBF | 500 | 0.1 | 0.1 | 0.84283 | 0.68791 | 0.10725 | 0.02935 |
| RBF | 1000 | 0.1 | 0.1 | 0.86679 | 0.65473 | 0.10707 | 0.04700 |
| RBF | 500 | 0.1 | 1.0 | 0.87642 | 0.64487 | 0.11213 | 0.08539 |
| RBF | 1000 | 0.1 | 1.0 | 0.86914 | 0.64779 | 0.09686 | 0.04042 |
| RBF | 500 | 0.1 | 10.0 | 0.83817 | 0.73330 | 0.10363 | 0.08562 |
| RBF | 1000 | 0.1 | 10.0 | 0.87773 | 0.62982 | 0.08651 | 0.01935 |
| RBF | 500 | 1.0 | 0.1 | 0.85124 | 0.68034 | 0.10533 | 0.07726 |
| RBF | 1000 | 1.0 | 0.1 | 0.88125 | 0.60208 | 0.09464 | 0.02572 |
| RBF | 500 | 1.0 | 1.0 | 0.85462 | 0.67836 | 0.10975 | 0.08118 |
| RBF | 1000 | 1.0 | 1.0 | 0.88768 | 0.59401 | 0.09479 | 0.04439 |
| RBF | 500 | 1.0 | 10.0 | 0.85299 | 0.66042 | 0.10038 | 0.05158 |
| RBF | 1000 | 1.0 | 10.0 | 0.86256 | 0.66650 | 0.09453 | 0.05881 |
| RBF | 500 | 10.0 | 0.1 | 0.85599 | 0.65100 | 0.09348 | 0.06703 |
| RBF | 1000 | 10.0 | 0.1 | 0.88294 | 0.60353 | 0.09154 | 0.02451 |
| RBF | 500 | 10.0 | 1.0 | 0.83700 | 0.72030 | 0.09499 | 0.06603 |
| RBF | 1000 | 10.0 | 1.0 | 0.88938 | 0.59334 | 0.09290 | 0.04248 |
| RBF | 500 | 10.0 | 10.0 | 0.85359 | 0.70931 | 0.08560 | 0.04297 |
| RBF | 1000 | 10.0 | 10.0 | 0.88089 | 0.61582 | 0.09105 | 0.03352 |

Error observed using RBF kernels with different combination(length scale and variance), R2 score, RMSE, MNLL and standard RMSE are used. For error quantification best results are highlighted.
In this table we observe that RBF is fairly a good kernel for this problem. We choose 500 and 1000 sample points to start with and divide it into 80-20 percent. We also have a very similar result on varying the length scale and variance, hence not much is achieved while changing the parameters over a factor of 10, However we are using m.optimize, hence it automatically adjusts the values and much effect of the parameters cant be observed. Overall, if more samples are considered, RBF could reach to a very good fit, but will be costly. A GPU will be needed each time to fit it.

| kernel name | pts | ls | var | R2 score | RMSE | MNLL | Std RMSE |
|---|---|---|---|---|---|---|---|
| Exp | 500 | 0.1 | 0.1 | 0.82431 | 0.71470 | 0.28748 | 0.04010 |
| Exp | 1000 | 0.1 | 0.1 | 0.88872 | 0.60506 | 0.30230 | 0.03777 |
| Exp | 500 | 0.1 | 1.0 | 0.87777 | 0.64489 | 0.29223 | 0.05768 |
| Exp | 1000 | 0.1 | 1.0 | 0.88079 | 0.61486 | 0.30018 | 0.03161 |
| Exp | 500 | 0.1 | 10.0 | 0.84547 | 0.69000 | 0.29714 | 0.04264 |
| Exp | 1000 | 0.1 | 10.0 | 0.90946 | 0.52262 | 0.30433 | 0.03619 |
| Exp | 500 | 1.0 | 0.1 | 0.88900 | 0.59132 | 0.30543 | 0.05930 |
| Exp | 1000 | 1.0 | 0.1 | 0.89097 | 0.58174 | 0.30392 | 0.04903 |
| Exp | 500 | 1.0 | 1.0 | 0.86709 | 0.62865 | 0.30971 | 0.06908 |
| Exp | 1000 | 1.0 | 1.0 | 0.88941 | 0.60661 | 0.30046 | 0.04704 |
| Exp | 500 | 1.0 | 10.0 | 0.86992 | 0.64896 | 0.29768 | 0.04262 |
| Exp | 1000 | 1.0 | 10.0 | 0.89503 | 0.59042 | 0.30361 | 0.03667 |
| Exp | 500 | 10.0 | 0.1 | 0.87024 | 0.62915 | 0.29696 | 0.04428 |
| Exp | 1000 | 10.0 | 0.1 | 0.90925 | 0.54663 | 0.30159 | 0.01673 |
| Exp | 500 | 10.0 | 1.0 | 0.85649 | 0.65917 | 0.29600 | 0.04515 |
| Exp | 1000 | 10.0 | 1.0 | 0.87394 | 0.62449 | 0.29330 | 0.03062 |
| Exp | 500 | 10.0 | 10.0 | 0.84426 | 0.70316 | 0.29287 | 0.06494 |
| Exp | 1000 | 10.0 | 10.0 | 0.89711 | 0.57742 | 0.29386 | 0.02796 |

Error observed using Exponential kernels with different combination(lengthscale and variance), R2 score, RMSE, MNLL and standard RMSE are used. For error quantification best results are highlighted.

Exponential kernel gives better results than the RBF kernel, we are judging better results by R2 score, which is the most recognized error estimate for regression problems. There is no desired value of R2 score to be reached but a score closer to one gives the user a confidence that his fit is good. This case also has the same effects of parameters as the RBF case.

| kernel name | pts | ls | var | R2 score | RMSE | MNLL | Std RMSE |
|---|---|---|---|---|---|---|---|
| Mat32 | 500 | 0.1 | 0.1 | 0.86166 | 0.65756 | 0.21210 | 0.04351 |
| Mat32 | 1000 | 0.1 | 0.1 | 0.88871 | 0.60455 | 0.19457 | 0.02393 |
| Mat32 | 500 | 0.1 | 1.0 | 0.86305 | 0.65934 | 0.21744 | 0.06009 |
| Mat32 | 1000 | 0.1 | 1.0 | 0.90046 | 0.56331 | 0.20948 | 0.02700 |
| Mat32 | 500 | 0.1 | 10.0 | 0.86772 | 0.62160 | 0.21194 | 0.06963 |
| Mat32 | 1000 | 0.1 | 10.0 | 0.90556 | 0.56019 | 0.21430 | 0.03279 |
| Mat32 | 500 | 1.0 | 0.1 | 0.83532 | 0.72865 | 0.19824 | 0.09471 |
| Mat32 | 1000 | 1.0 | 0.1 | 0.90164 | 0.55424 | 0.22347 | 0.06132 |
| Mat32 | 500 | 1.0 | 1.0 | 0.85627 | 0.64921 | 0.19143 | 0.04993 |
| Mat32 | 1000 | 1.0 | 1.0 | 0.89432 | 0.58666 | 0.20902 | 0.05284 |
| Mat32 | 500 | 1.0 | 10.0 | 0.87433 | 0.63001 | 0.21923 | 0.04802 |
| Mat32 | 1000 | 1.0 | 10.0 | 0.87292 | 0.64179 | 0.20135 | 0.03977 |
| Mat32 | 500 | 10.0 | 0.1 | 0.86575 | 0.63491 | 0.21152 | 0.08523 |
| Mat32 | 1000 | 10.0 | 0.1 | 0.89049 | 0.58283 | 0.22233 | 0.03749 |
| Mat32 | 500 | 10.0 | 1.0 | 0.86069 | 0.64932 | 0.21969 | 0.05768 |
| Mat32 | 1000 | 10.0 | 1.0 | 0.88311 | 0.62394 | 0.20971 | 0.04286 |
| Mat32 | 500 | 10.0 | 10.0 | 0.83698 | 0.70185 | 0.20869 | 0.03379 |
| Mat32 | 1000 | 10.0 | 10.0 | 0.89391 | 0.58531 | 0.20781 | 0.05013 |

Error observed using RBF kernels with different combination(lengthscale and variance), Matern32 score, RMSE, MNLL and standard RMSE are used. For error quantification best results are highlighted.

Matern32 kernel gives good values of fit at low length scale combined with a high variance, the standard root mean square error also follows the desired property that it should be less than the RMSE. We also have a property that fit on 1000 points is better than the fit on 500 points, which means the choice of the kernel is suitable and the fit is extremely good at most of the places. However there is not a sharp rise in the R2 score when points are doubled, it means there are some areas where data is not approximated well.

| kernel name | pts | ls | var | R2 score | RMSE | MNLL | Std RMSE |
|---|---|---|---|---|---|---|---|
| Mat52 | 500 | 0.1 | 0.1 | 0.85883 | 0.66092 | 0.17642 | 0.06594 |
| Mat52 | 1000 | 0.1 | 0.1 | 0.90758 | 0.56094 | 0.18016 | 0.06769 |
| Mat52 | 500 | 0.1 | 1.0 | 0.85830 | 0.65612 | 0.16428 | 0.02897 |
| Mat52 | 1000 | 0.1 | 1.0 | 0.88556 | 0.61102 | 0.16301 | 0.02219 |
| Mat52 | 500 | 0.1 | 10.0 | 0.85859 | 0.68046 | 0.15526 | 0.07536 |
| Mat52 | 1000 | 0.1 | 10.0 | 0.88699 | 0.59882 | 0.15444 | 0.03674 |
| Mat52 | 500 | 1.0 | 0.1 | 0.87779 | 0.62298 | 0.17951 | 0.06358 |
| Mat52 | 1000 | 1.0 | 0.1 | 0.86939 | 0.64044 | 0.17554 | 0.03485 |
| Mat52 | 500 | 1.0 | 1.0 | 0.88054 | 0.58881 | 0.18552 | 0.11465 |
| Mat52 | 1000 | 1.0 | 1.0 | 0.90714 | 0.55446 | 0.17467 | 0.02174 |
| Mat52 | 500 | 1.0 | 10.0 | 0.86197 | 0.66137 | 0.19046 | 0.03779 |
| Mat52 | 1000 | 1.0 | 10.0 | 0.88754 | 0.58193 | 0.16701 | 0.03415 |
| Mat52 | 500 | 10.0 | 0.1 | 0.83404 | 0.71988 | 0.16444 | 0.06412 |
| Mat52 | 1000 | 10.0 | 0.1 | 0.89584 | 0.57779 | 0.16668 | 0.02818 |
| Mat52 | 500 | 10.0 | 1.0 | 0.85539 | 0.66901 | 0.18099 | 0.02863 |
| Mat52 | 1000 | 10.0 | 1.0 | 0.89045 | 0.58911 | 0.17434 | 0.03218 |
| Mat52 | 500 | 10.0 | 10.0 | 0.85185 | 0.68602 | 0.17482 | 0.11884 |
| Mat52 | 1000 | 10.0 | 10.0 | 0.89983 | 0.57698 | 0.17536 | 0.03322 |

Error observed using Matern52 kernels with different combination(lengthscale and variance), R2 score, RMSE, MNLL and standard RMSE are used. For error quantification best results are highlighted.
Matern52 shows a remarkable difference with the Matern32, that is the results are best when both the lengthscale and variance is low. This means less peaks are required by the second order differentiable function to approximate the booster function whereas more peaks are required by the first order differentiable function to fit the data.

| kernel name | pts | ls | var | R2 score | RMSE | MNLL | Std RMSE |
|---|---|---|---|---|---|---|---|
| Exp+M32 | 500 | 0.1 | 0.1 | 0.87786 | 0.62810 | 0.25175 | 0.06096 |
| Exp+M32 | 1000 | 0.1 | 0.1 | 0.87846 | 0.63743 | 0.27642 | 0.02544 |
| Exp+M32 | 500 | 0.1 | 1.0 | 0.86321 | 0.66816 | 0.27566 | 0.03040 |
| Exp+M32 | 1000 | 0.1 | 1.0 | 0.89669 | 0.58483 | 0.27844 | 0.07460 |
| Exp+M32 | 500 | 0.1 | 10.0 | 0.88650 | 0.62016 | 0.26595 | 0.02690 |
| Exp+M32 | 1000 | 0.1 | 10.0 | 0.87976 | 0.59945 | 0.28732 | 0.04420 |
| Exp+M32 | 500 | 1.0 | 0.1 | 0.87247 | 0.64634 | 0.28153 | 0.05008 |
| Exp+M32 | 1000 | 1.0 | 0.1 | 0.90047 | 0.56983 | 0.29635 | 0.03777 |
| Exp+M32 | 500 | 1.0 | 1.0 | 0.88732 | 0.59734 | 0.29494 | 0.04516 |
| Exp+M32 | 1000 | 1.0 | 1.0 | 0.90639 | 0.54002 | 0.30186 | 0.02720 |
| Exp+M32 | 500 | 1.0 | 10.0 | 0.84619 | 0.68768 | 0.28074 | 0.03885 |
| Exp+M32 | 1000 | 1.0 | 10.0 | 0.88921 | 0.59995 | 0.28687 | 0.02503 |
| Exp+M32 | 500 | 10.0 | 0.1 | 0.86520 | 0.66791 | 0.24997 | 0.03580 |
| Exp+M32 | 1000 | 10.0 | 0.1 | 0.89966 | 0.55933 | 0.22547 | 0.04413 |
| Exp+M32 | 500 | 10.0 | 1.0 | 0.84432 | 0.69011 | 0.22758 | 0.05412 |
| Exp+M32 | 1000 | 10.0 | 1.0 | 0.87878 | 0.63902 | 0.22672 | 0.02939 |
| Exp+M32 | 500 | 10.0 | 10.0 | 0.85890 | 0.66873 | 0.21989 | 0.08071 |
| Exp+M32 | 1000 | 10.0 | 10.0 | 0.91559 | 0.53117 | 0.22687 | 0.04395 |

Error observed using Exponential+M32 kernels with different combination(lengthscale and variance), R2 score, RMSE, MNLL and standard RMSE are used. For error quantification best results are highlighted.
We now try a guess to add the Exponential kernel and the Matern32 function, as we observed we need a high variance to fit the data, and we get the same result. A high lengthscale is also neede in this case because of the exponential function.

| kernel name | pts | ls | var | R2 score | RMSE | MNLL | Std RMSE |
|---|---|---|---|---|---|---|---|
| M52*RBF | 500 | 0.1 | 0.1 | 0.84811 | 0.67704 | 0.18697 | 0.06817 |
| M52*RBF | 1000 | 0.1 | 0.1 | 0.88086 | 0.61822 | 0.17519 | 0.05866 |
| M52*RBF | 500 | 0.1 | 1.0 | 0.84911 | 0.68811 | 0.17345 | 0.05219 |
| M52*RBF | 1000 | 0.1 | 1.0 | 0.86920 | 0.66004 | 0.15261 | 0.02731 |
| M52*RBF | 500 | 0.1 | 10.0 | 0.88082 | 0.60672 | 0.17307 | 0.06903 |
| M52*RBF | 1000 | 0.1 | 10.0 | 0.89026 | 0.60922 | 0.16295 | 0.03251 |
| M52*RBF | 500 | 1.0 | 0.1 | 0.87882 | 0.63020 | 0.18200 | 0.04477 |
| M52*RBF | 1000 | 1.0 | 0.1 | 0.90353 | 0.56771 | 0.17681 | 0.03259 |
| M52*RBF | 500 | 1.0 | 1.0 | 0.86653 | 0.65585 | 0.17483 | 0.02630 |
| M52*RBF | 1000 | 1.0 | 1.0 | 0.90669 | 0.54171 | 0.17949 | 0.05095 |
| M52*RBF | 500 | 1.0 | 10.0 | 0.89088 | 0.59123 | 0.17907 | 0.04817 |
| M52*RBF | 1000 | 1.0 | 10.0 | 0.90452 | 0.54962 | 0.16507 | 0.04630 |
| M52*RBF | 500 | 10.0 | 0.1 | 0.86159 | 0.66368 | 0.18149 | 0.05352 |
| M52*RBF | 1000 | 10.0 | 0.1 | 0.89130 | 0.59609 | 0.16376 | 0.01426 |
| M52*RBF | 500 | 10.0 | 1.0 | 0.86956 | 0.64317 | 0.17958 | 0.08067 |
| M52*RBF | 1000 | 10.0 | 1.0 | 0.87805 | 0.62143 | 0.17710 | 0.06333 |
| M52*RBF | 500 | 10.0 | 10.0 | 0.86213 | 0.66745 | 0.15652 | 0.07453 |
| M52*RBF | 1000 | 10.0 | 10.0 | 0.89591 | 0.58045 | 0.17955 | 0.02549 |

Error observed using M52*RBF kernels with different combination(lengthscale and variance), R2 score, RMSE, MNLL and standard RMSE are used. For error quantification best results are highlighted.

We now try a new kernel Matern52 product with RBF, as we can make a guess thatlow variance as compared to the Matern32 value should be good with this, it comes out to be true. Very good values of R2 score are observed in general.

| kernel name | pts | ls | var | R2 score | RMSE | MNLL | Std RMSE |
|---|---|---|---|---|---|---|---|
| Exp+RBF | 500 | 0.1 | 0.1 | 0.85159 | 0.67709 | 0.27678 | 0.04326 |
| Exp+RBF | 1000 | 0.1 | 0.1 | 0.90152 | 0.55204 | 0.29381 | 0.03203 |
| Exp+RBF | 500 | 0.1 | 1.0 | 0.86841 | 0.64169 | 0.29044 | 0.04605 |
| Exp+RBF | 1000 | 0.1 | 1.0 | 0.88193 | 0.59507 | 0.29184 | 0.04891 |
| Exp+RBF | 500 | 0.1 | 10.0 | 0.86999 | 0.61748 | 0.29482 | 0.03867 |
| Exp+RBF | 1000 | 0.1 | 10.0 | 0.89485 | 0.59442 | 0.29006 | 0.04349 |
| Exp+RBF | 500 | 1.0 | 0.1 | 0.89710 | 0.57461 | 0.28661 | 0.04112 |
| Exp+RBF | 1000 | 1.0 | 0.1 | 0.88224 | 0.61559 | 0.29722 | 0.01925 |
| Exp+RBF | 500 | 1.0 | 1.0 | 0.87219 | 0.65022 | 0.26917 | 0.05402 |
| Exp+RBF | 1000 | 1.0 | 1.0 | 0.88494 | 0.60293 | 0.29361 | 0.01369 |
| Exp+RBF | 500 | 1.0 | 10.0 | 0.88073 | 0.59334 | 0.28472 | 0.05993 |
| Exp+RBF | 1000 | 1.0 | 10.0 | 0.89834 | 0.57735 | 0.28273 | 0.04122 |
| Exp+RBF | 500 | 10.0 | 0.1 | 0.87260 | 0.63284 | 0.27844 | 0.07450 |
| Exp+RBF | 1000 | 10.0 | 0.1 | 0.89014 | 0.59406 | 0.28919 | 0.05952 |
| Exp+RBF | 500 | 10.0 | 1.0 | 0.84972 | 0.66180 | 0.27939 | 0.05610 |
| Exp+RBF | 1000 | 10.0 | 1.0 | 0.88655 | 0.60151 | 0.28508 | 0.04149 |
| Exp+RBF | 500 | 10.0 | 10.0 | 0.83283 | 0.69272 | 0.28705 | 0.03904 |
| Exp+RBF | 1000 | 10.0 | 10.0 | 0.88838 | 0.60101 | 0.28705 | 0.03641 |

Error observed using Exponential+RBF kernels with different combination(lengthscale and variance), R2 score, RMSE, MNLL and standard RMSE are used. For error quantification best results are highlighted.
We now try a new kernel exponential addition with RBF, as we can make a guess thatlow variance as compared to the Matern32 value should be good with this, it comes out to be true. Very good values of R2 score are observed in general.

| kernel name | pts | ls | var | R2 score | RMSE | MNLL | Std RMSE |
|---|---|---|---|---|---|---|---|
| RBF | 1000 | 10.0 | 1.0 | 0.88938 | 0.59334 | 0.09290 | 0.04248 |
| Exp | 1000 | 10.0 | 0.1 | 0.90925 | 0.54663 | 0.30159 | 0.01673 |
| Mat32 | 1000 | 0.1 | 10.0 | 0.90556 | 0.56019 | 0.21430 | 0.03279 |
| Mat52 | 1000 | 0.1 | 0.1 | 0.90758 | 0.56094 | 0.18016 | 0.06769 |
| M52*RBF | 1000 | 1.0 | 10.0 | 0.90452 | 0.54962 | 0.16507 | 0.04630 |
| Exp+M32 | 1000 | 10.0 | 10.0 | 0.91559 | 0.53117 | 0.22687 | 0.04395 |
| Exp+RBF | 1000 | 0.1 | 0.1 | 0.90152 | 0.55204 | 0.29381 | 0.03203 |

The best from the table is Exp+M32
This table shows all the best possible combinations of above kernels and from that we observed that the best configurations are lengthscale and variance is 10 for Exp+M32.

# 2 Part 2 BO

In this part we have to optimize the booster function to calculate its optima, optimum values of booster function are tuff to calculate because the function is a black box function and is not easy to compute, only selected values are tranfered to the function to calculate the value over them as explicit calculation at all points is not possible. We use 5 acquisition function of Bayesian optimisation to calculate the optimum values of the function. Acquisition functions are mathematical techniques that guide how the parameter space should be explored during Bayesian optimization. They use the predicted mean and predicted variance generated by the Gaussian process model. The predicted variance is very close to zero at or very near to an existing result. The first three tables of Probability improvement, UCB, direct approach are not considered constraints and rest two cases consider constraints.

| iter | x1 | x2 | x3 | x4 | y |
|---|---|---|---|---|---|
| 10 | 0.288 | 0.016 | 0.322 | 0.891 | 4.536 |
| 20 | 0.727 | 0.513 | 0.164 | 0.957 | 4.820 |
| 30 | 0.357 | 0.936 | 0.403 | 0.962 | 5.114 |
| 40 | 0.425 | 0.869 | 0.176 | 0.711 | 5.102 |
| 50 | 0.468 | 0.751 | 0.086 | 0.775 | 5.208 |
| 60 | 0.486 | 0.670 | 0.337 | 0.973 | 5.038 |

EI

The first approach we use is the Expected Improvement approach, it is one of the most used approaches of Bayesian optimization to calculate the optima. We perform the calculation on 20 points and add upto 60 points in the sample space. The results seem to converge near 5 and indicate the convergence. The convergence plots of part 2 can be found with the attached file and they show convergence.

| iter | x1 | x2 | x3 | x4 | y |
|---|---|---|---|---|---|
| 10 | 0.336 | 0.713 | 0.107 | 0.779 | 5.181 |
| 20 | 0.227 | 0.657 | 0.167 | 0.746 | 5.083 |
| 30 | 0.426 | 0.949 | 0.323 | 0.948 | 5.212 |
| 40 | 0.427 | 0.250 | 0.163 | 0.986 | 5.099 |
| 50 | 0.879 | 0.966 | 0.069 | 0.778 | 5.055 |
| 60 | 0.189 | 0.413 | 0.273 | 0.915 | 5.265 |

PI

The second approach we use is the probability improvement, In this case also, the convergence value is near to the one in case of EI, which solidifies our belief that the results are robust.

| iter | x1 | x2 | x3 | x4 | y |
|---|---|---|---|---|---|
| 10 | 0.215 | 0.332 | 0.120 | 0.756 | 5.196 |
| 20 | 0.157 | 0.659 | 0.268 | 0.906 | 5.175 |
| 30 | 0.221 | 0.344 | 0.366 | 0.811 | 4.897 |
| 40 | 0.314 | 0.674 | 0.234 | 0.917 | 5.324 |
| 50 | 0.218 | 0.418 | 0.194 | 0.999 | 5.480 |
| 60 | 0.985 | 0.998 | 0.114 | 0.974 | 5.143 |

UCB

The third approach we use is the UCB, In this case also, the convergence value is near to the one in case of EI, which solidifies our belief that the results are robust.

| iter | x1 | x2 | x3 | x4 | y |
|---|---|---|---|---|---|
| 10 | 0.262 | 0.283 | 0.118 | 0.889 | 5.367 |
| 20 | 0.217 | 0.413 | 0.202 | 0.907 | 5.353 |
| 30 | 0.483 | 0.779 | 0.135 | 0.772 | 5.210 |
| 40 | 0.374 | 0.675 | 0.253 | 0.911 | 5.254 |
| 50 | 0.180 | 0.211 | 0.181 | 0.786 | 5.193 |
| 60 | 0.171 | 0.468 | 0.177 | 0.784 | 5.157 |

Direct approach

The fourth approach we use is the direct approach, In this case also, the convergence value is near to the one in case of EI, which solidifies our belief that the results are robust.

| iter | x1 | x2 | x3 | x4 | y |
|---|---|---|---|---|---|
| 10 | 0.328 | 0.615 | 0.402 | 1.000 | 5.093 |
| 20 | 0.264 | 0.389 | 0.152 | 0.995 | 5.503 |
| 30 | 0.801 | 0.889 | 0.081 | 0.904 | 5.236 |
| 40 | 0.433 | 0.920 | 0.121 | 0.909 | 5.383 |
| 50 | 0.597 | 0.882 | 0.261 | 0.967 | 5.201 |
| 60 | 0.220 | 0.161 | 0.026 | 0.935 | 5.140 |

POF The fifth approach we use is POF, In this case also, the convergence value is near to the one in case of EI, which solidifies our belief that the results are robust.