Problem 1 report:

• The linear program for the general problem written as an objective and set of constraints

Objective Function:

Minimize t

This represents the minimization of the maximum absolute deviation of the regression line from the data points.

Constraints:

For each data point (xi,yi), the deviation must satisfy: |a * xi + b − yi| <= t

Equivalently, we can express this as two constraints for every point:

a * xi + b - yi <= t and - (a * xi + b - yi) <= t

Variables:

a and b: Parameters of the regression line y = ax + b.

t: The maximum absolute deviation (error) across all data points.

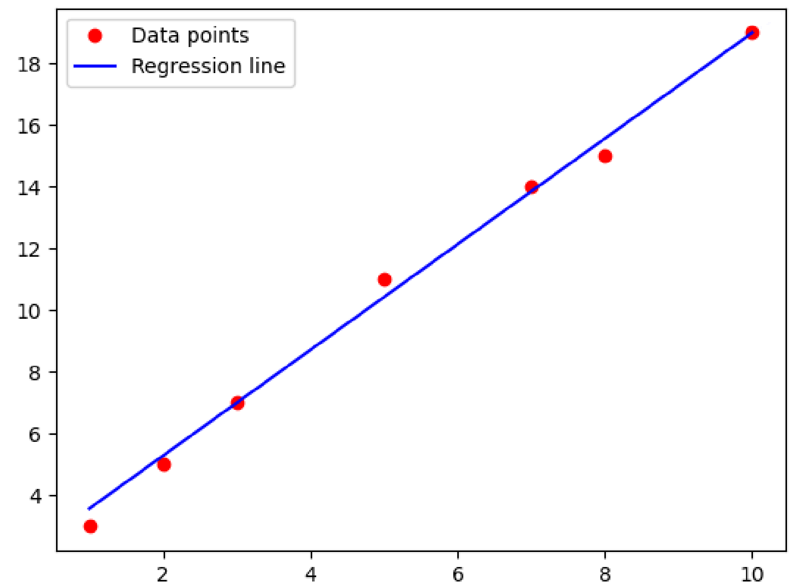
•The best solution for the specific problem above

```
Optimal a: 1.7142857144575163
Optimal b: 1.8571428563570866
Optimal t (max absolute deviation): 0.5714285660084366
```

•The output of the LP solver that you used (showing that an optimal solution was found)

```
constraints = [cp.abs(a * x_val[i] + b - y_val[i]) <= t for i in range(len(points))]

objective = cp.Minimize(t)
prob = cp.Problem(objective, constraints)
prob.solve()
```

•A plot of the points and your solution for the instance



Problem 2 report:

•A description for a linear program for finding the best fit curve for temperature data.

Decision Variables:

- x0,x1,x2,x3,x4,x5: Coefficients of the model.
- t: The maximum absolute deviation (error) between the model prediction and the observed temperature.

Objective Function:

Minimize t

This minimizes the worst-case (maximum) absolute error across all data points.

Constraints:

For each data point i: -t <= Tpred(di) - Ti <= t where Ti is the observed temperature for day di.

•The values of all of the variables to your linear program in the optimal solution that your linear program solver finds for the Corvallis data. Solving this LP may take a while depending on your computer. Include the output of the LP solver that you use (showing that an optimal solution was found).

```
Optimal solution found:
x0 = 8.021419721336493
x1 = 0.00010694836293243726  (daily drift in °C)
x2 = 4.280890747505641
x3 = 8.186857819354795
x4 = −0.7906307941032018
x5 = −0.2953602081034133
Minimum maximum absolute deviation (E) = 14.235540295367645
Estimated annual drift (x1 * 365.25) = 0.039062889561072706 °C/year
```

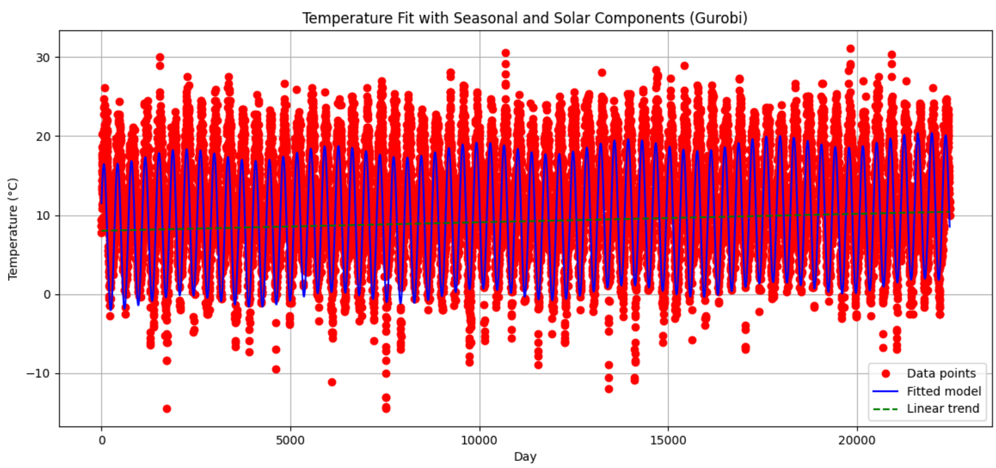
•A single plot that contains:

–the raw data plotted as points in 2-d (with d as the x-axis and T as the y-axis),

–your best fit curve, and

–the linear part of the curve x0 + x1·d.

```
Thread count: 8 physical cores, 8 logical processors, using up to 8 threads

Optimize a model with 44608 rows, 7 columns and 312220 nonzeros
Model fingerprint: 0xc0d516ac
Coefficient statistics:
  Matrix range     [6e-05, 2e+04]
  Objective range  [1e+00, 1e+00]
  Bounds range     [0e+00, 0e+00]
  RHS range        [2e-01, 3e+01]
Presolve time: 0.07s
Presolved: 7 rows, 44608 columns, 312220 nonzeros

Iteration    Objective       Primal Inf.    Dual Inf.      Time
       0    7.0015891e+33   1.726236e+34   7.001589e+03      0s

Starting sifting (using simplex for sub-problems)...

    Iter     Pivots    Primal Obj      Dual Obj        Time
       0         19    -infinity      1.9194274e+01      0s
       1         30   -0.0000000e+00  1.7613099e+01      0s
       2         48    1.1961771e+01  1.6295135e+01      0s

Sifting complete

      64    1.4235540e+01   0.000000e+00   0.000000e+00      0s

Solved in 64 iterations and 0.11 seconds (0.28 work units)
Optimal objective  1.423554030e+01
Optimal solution found:
```



•Based on the value x1, how many degrees Celsius per century is Corvallis changing, and is it warming or cooling trend?

Multiply x1 by 365.25 * 100 => temp_change_per_century = x_values[1] * 365.25 * 100

If temp_change_per_century is positive, the trend is warming.

If negative, it indicates a cooling trend.