Assignment 1

Machine Learning, SS2021

Team members		
Last name	First name	Matriculation Number
Rodiga	Luca	1620972
David	Tamara	11813320

1.1.A

$$\frac{\partial E}{\partial b} \sum_{i=1}^{m} (y_i - (ax_i + b))^2 = -2 \sum_{i=1}^{m} y_i - ax_i - b$$

$$-\cancel{2} \sum_{i=1}^{m} y_i - ax_i - b = 0$$

$$\sum_{i=1}^{m} b = \sum_{i=1}^{m} y_i - ax_i$$

$$mb = \sum_{i=1}^{m} y_i - ax_i$$

$$b = \frac{1}{m} \sum_{i=1}^{m} y_i - ax_i$$

$$b = \bar{y} - a\bar{x}$$

$$\frac{\partial E}{\partial a} \sum_{i=1}^{m} (y_i - (ax_i + b))^2 = -2 \sum_{i=1}^{m} x_i (y_i - (ax_i + b))$$

$$-\cancel{2} \sum_{i=1}^{m} x_i (y_i - ax_i - b) = 0$$

$$\sum_{i=1}^{m} x_i y_i - ax_i^2 - bx_i = 0$$

$$\sum x_i y_i - ax_i^2 - (\bar{y} - a\bar{x}) x_i = 0$$

$$\sum x_i y_i - ax_i^2 - x_i \bar{y} + a\bar{x} x_i = 0$$

$$\sum (y_i - ax_i - \bar{y} + a\bar{x}) x_i = 0$$

$$\sum y_i - \bar{y} + a(\bar{x} - x_i) = 0$$

$$\sum y_i - \bar{y} = -a \sum \bar{x} - x_i$$

$$a = \frac{\sum y_i - \bar{y}}{\sum x_i - \bar{x}}$$

$$a = \frac{\sum (y_i - \bar{y}) \cdot (x_i - \bar{x})}{\sum (x_i - \bar{x}) \cdot (x_i - \bar{x})}$$

$$a = \frac{\sum_{i=1}^{m} (y_i - \bar{y}) \cdot (x_i - \bar{x})}{\sum_{i=1}^{m} (x_i - \bar{x})^2} = \frac{cov(x, y)}{var(x)}$$

Intersection

Freitag, 16. April 2021 09:55

$y = ax + b$

$y = cx + d$

$$ax + b = cx + d$$

$$ax - cx = d - b$$

$$x \cdot (a - c) = d - b$$

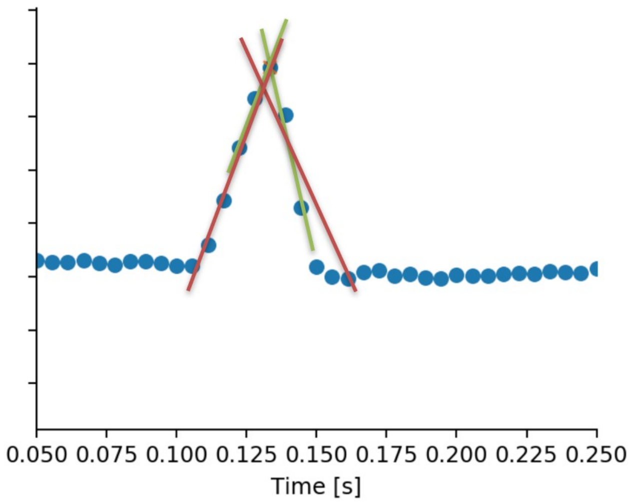
$$x = \frac{d - b}{a - c}$$

1.5

The first condition checks, if the new peak has a higher value/amplitude than the old peak. So, in short it checks, if the new peak is higher than the old peak. The second condition constrains the timeframe to one time step before and one time step after the old peak to check, if really a new peak was found and not a subsequent and already existing peak.

1.6

The sketch below illustrates, how new lines could be fitted to a peak. The green lines were approximately generated with 3 data points each, while the red lines were generated with 6 data points. It clearly shows that taking the fifth and sixth datapoint into account, shifts the right, red line away from the peak, which leads to a worse result of the intersection. Thus, calculating the regression line with too many data points, leads to worse results, because data points, which are not part of the peak, are taken into account.



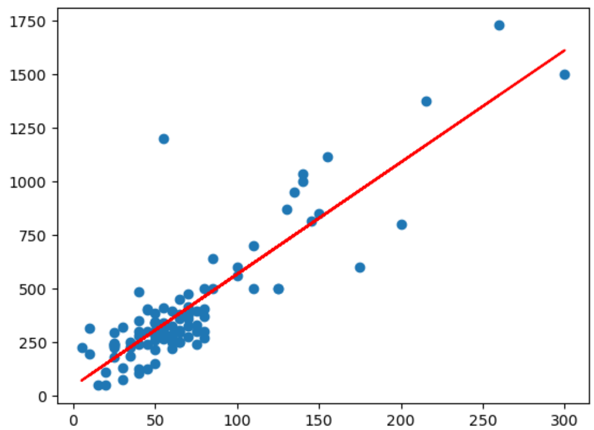
1.1.7

3 data points including the peak to fit both the right and left line returns the best result, which was:
Improved peaks: 80.0,
total peaks: 83
Percentage of peaks improved: 0.9639

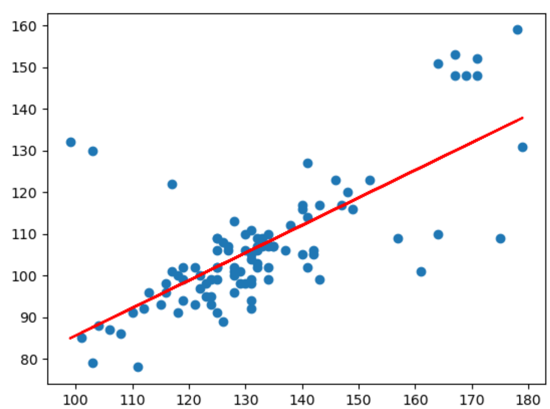
1.1.8
Because parabolas are always symmetric and, in this case, the data points are not symmetrically distributed around the peaks. Therefore, the approach with fitting lines to improve peaks and finding an intersection point is preferable here, because we don't depend on symmetrically distributed data points.

1.2

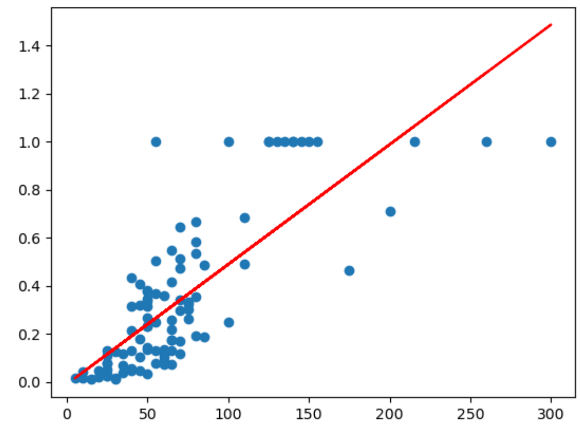
duration -> calories
pearson_coeff
0.8689894608314118
theta
[46.23868585 5.22109411]
mse
21843.026316801122

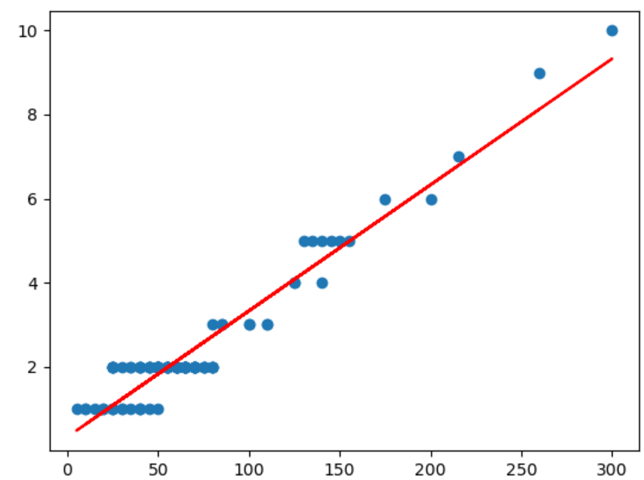


max_pulse -> avg_pulse
pearson_coeff
0.7182676942597552
theta
[19.39448609 0.66174078]
mse
122.72354176243724



duration -> exercise_intensity
pearson_coeff
0.7840105994885613
theta
[-0.00960533 0.00498852]
mse
0.0385509680784015

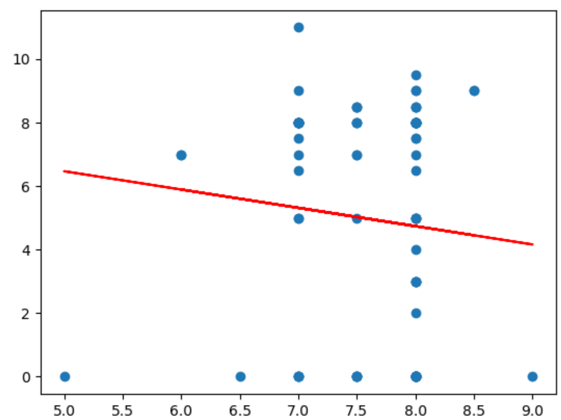




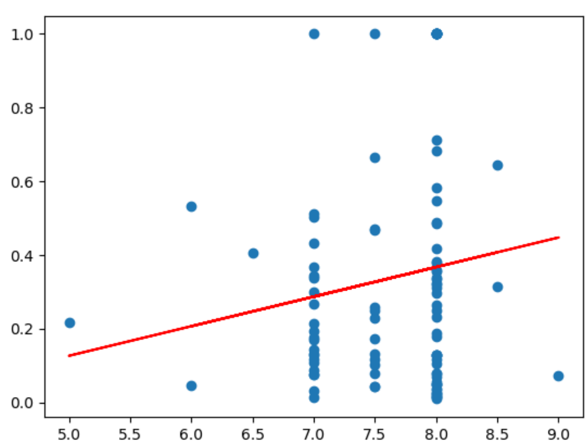
duration -> fitness_level
pearson_coeff
0.9582020994401034
theta
[0.32695313 0.03000784]
mse
0.19836859809302207

The plots show clearly a strong positive relation between the selected variables. Additionally, all Pearson coefficient are above 0.70, which also indicate a positive relation. In general values of the Pearson coefficient close to 1 and -1 indicate a strong positive or negative relation, while values close to 0 indicate that the variable are independent of each other.

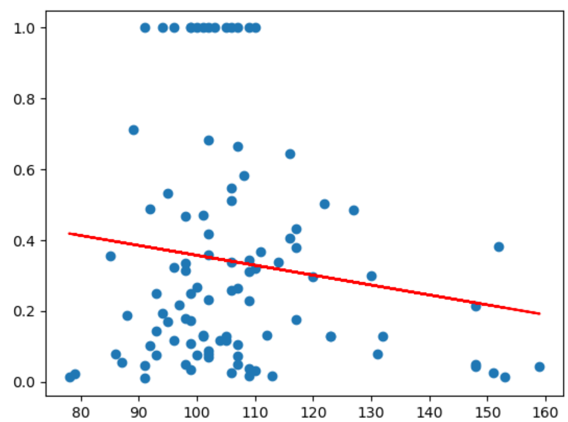
No linear relations
hours_sleep -> hours_work
pearson_coeff
-0.09107038009607227
theta
[9.35513227 -0.57696559]
mse
13.504558986073244



hours_sleep -> exercise_intensity
pearson_coeff
0.14769444488175465
theta
[-0.27400637 0.08020306]
mse
0.0978649137288099

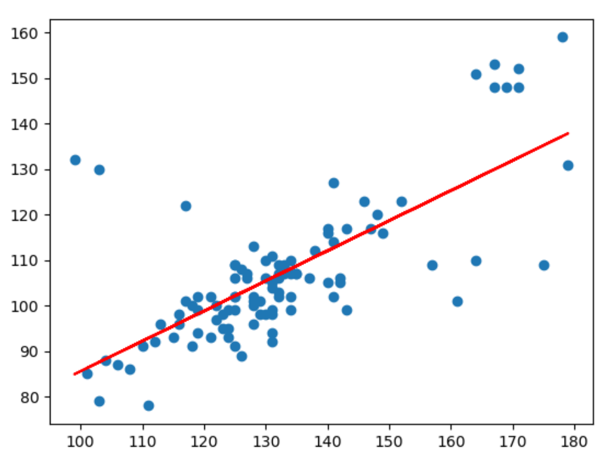


avg_pulse -> exercise_intensity
pearson_coeff
-0.1407044629330271
theta
[0.6370653 -0.00279518]
mse
0.09806659941846785



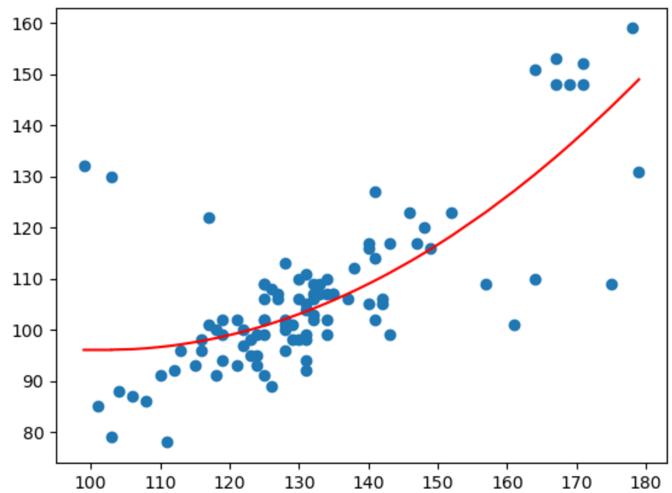
Polynomial regression
Polynomia degree 1l: max_pulse -> avg_pulse

theta
[19.39448609 0.66174078]
mse
122.72354176243724



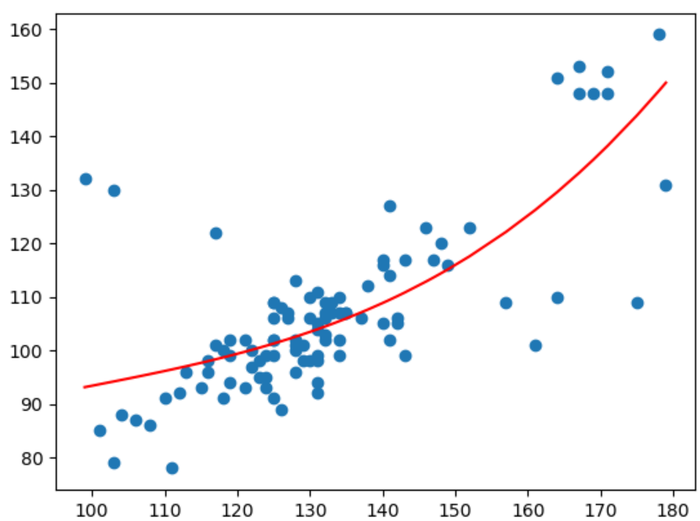
Polynomia degree 2l: max_pulse -> avg_pulse

theta
[1.88703058e+02 -1.81762297e+00 8.91462786e-03]
mse
108.97807235341814



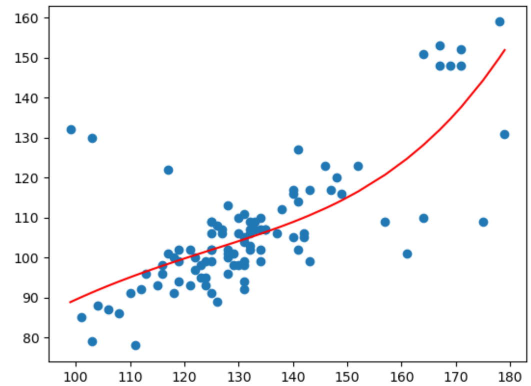
Polynomial degree 3: max_pulse -> avg_pulse

theta
[5.11966680e-02 2.29236239e+00 -2.05040562e-02 6.91477722e-05]
mse
111.51340490261816



Polynomial degree 4: max_pulse -> avg_pulse

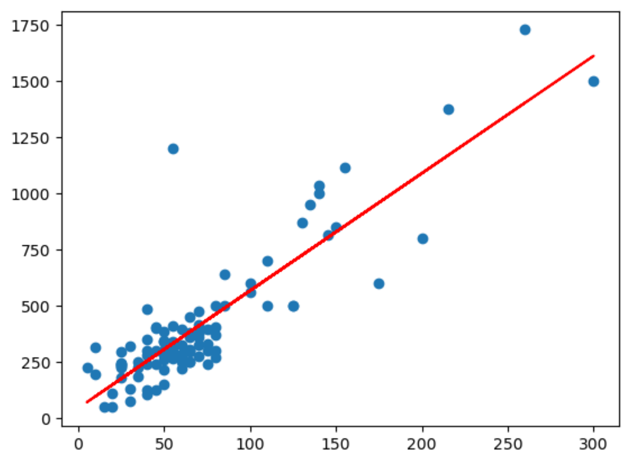
theta
[9.33533142e-06 6.30549827e-04 2.87285952e-02 -2.78748093e-04
  8.08490630e-07]
mse
116.72816450700246



The best polynomial regression of the relation max_pulse -> avg_pulse was with degree 2.

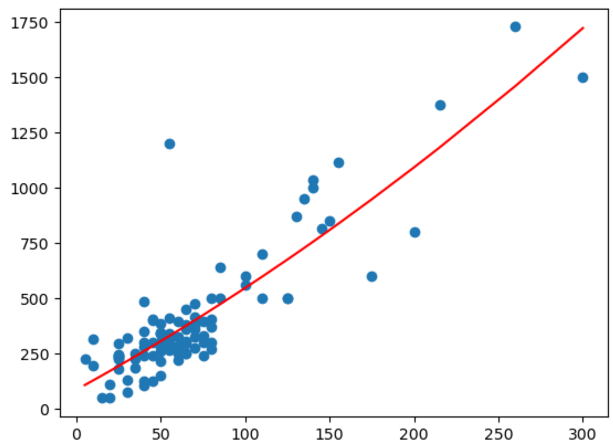
Polynomial degree 1: duration -> calories

theta
[46.23868585 5.22109411]
mse
21843.026316801122



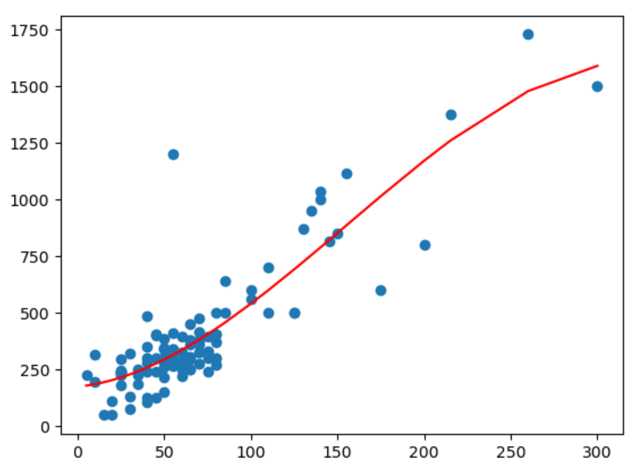
Polynomial degree 2: duration -> calories

theta
[8.78885057e+01 4.17786528e+00 4.24132094e-03]
mse
21498.68907560973



Polynomial degree 3: duration -> calories

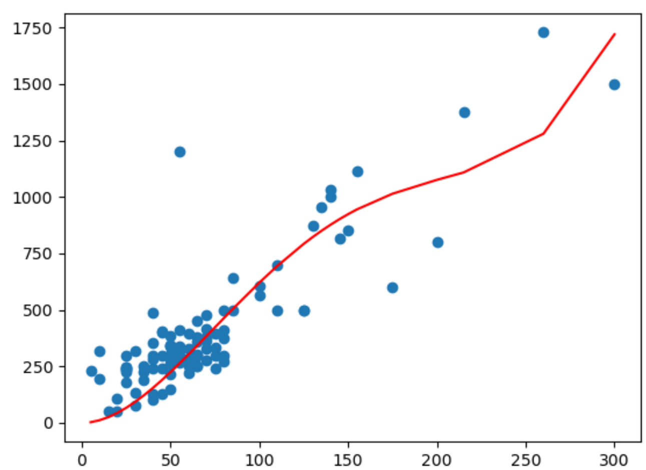
theta
[1.75402179e+02 7.46468217e-01 3.70554061e-02 -7.94132920e-05]
mse
20723.52254084419



Polynomial degree 4: duration -> calories

theta
[5.06941078e-05 2.82029494e-03 1.24690094e-01 -7.63148185e-04
  1.37066910e-06]
mse
29365.015555042588

The best polynomial regression of the relation duration -> calories was with degree 3.



Multilinear regression
duration & exercise_intensity & avg_pulse -> calories

theta
[-194.92286939   3.60904222 357.27688039   2.17759601]
mse
15394.685969930557

The multilinear regression improved the mse of duration -> calories from ~21843.03 to ~15394,69 with the relation duration & exercise_intensity & avg_pulse -> calories.

Task 2)
The best achieved test accuracy was 0.8525. Multiple combinations of different solvers and penalties achieved this accuracy. In the code are all combinations, which were tested with the corresponding results.
Below is the default configuration, which also achieved 0. 8525.

```
clf = LogisticRegression().fit(x_train, y_train)
```

Train accuracy: 0.8430. Test accuracy: 0.8525.
Train loss: 5.423523526218722. Test loss: 5.095963690799628.

Theta:

```
[[-0.07653786 -0.85844456  0.78464861 -0.1973674  -0.24089346 -0.13433343
   0.09092294  0.49998533 -0.4756523  -0.63029188  0.13306084 -0.8782799
  -0.45213954]]
```

Intercept:
[0.10624298]

2.4)
Logistic regression is used, when there are only 2 classifying classes required. Like, if the decision is like a boolean. The prediction can be either yes or no or 1 or 0.

2.5)

$$J(\theta) = \frac{1}{m}[\sum_{i=1}^{m} -y^{(i)} log(h_\theta(x^{(i)})) + (1 - y^{(i)}) log(1 - h_\theta(x^{(i)}))]$$

$$m = number\ of\ samples$$

Even if every y is predicted correctly, the loss function like above can only be 0, if there are exactly the same number of 1 and 0 zero predictions.

Task 3

3.3)

Ackley function: $\frac{d}{dx}\left[-20e^{-0.2\sqrt{0.5\cdot(x^2+y^2)}} - e^{0.5(\cos(2\pi x)+\cos(2\pi y))} + e + 20\right]$

$$= -\frac{d}{dx}\left[e^{\frac{\cos(2\pi x)+\cos(2\pi y)}{2}}\right] - 20\cdot\frac{d}{dx}\left[e^{-\frac{\sqrt{x^2+y^2}}{5\sqrt{2}}}\right] + \frac{d}{dx}[e] + \frac{d}{dx}[20]$$

$$= -e^{\frac{\cos(2\pi x)+\cos(2\pi y)}{2}} \cdot \frac{d}{dx}\left[\frac{\cos(2\pi x)+\cos(2\pi y)}{2}\right] - 20e^{\frac{-\sqrt{x^2+y^2}}{5\sqrt{2}}} \cdot \frac{d}{dx}\left[-\frac{\sqrt{x^2+y^2}}{5\sqrt{2}}\right] + 0 + 0$$

$$= -e^{\frac{\cos(2\pi x)+\cos(2\pi y)}{2}} \cdot \frac{1}{2}\left(\frac{d}{dx}[\cos(2\pi x)] + \frac{d}{dx}[\cos(2\pi y)]\right) - 20e^{\frac{-\sqrt{x^2+y^2}}{5\sqrt{2}}}\left(-\frac{1}{5\sqrt{2}}\cdot\frac{d}{dx}\left[\sqrt{x^2+y^2}\right]\right)$$

$$= -\frac{e^{\frac{\cos(2\pi x)+\cos(2\pi y)}{2}}\left((-\sin(2\pi x))\cdot\frac{d}{dx}[2\pi x]+0\right)}{2} + 2^{\frac{3}{2}}e^{-\frac{\sqrt{x^2+y^2}}{5\sqrt{2}}} \cdot \frac{1}{2}(x^2+y^2)^{\frac{1}{2}-1} \cdot \frac{d}{dx}[x^2+y^2]$$

$$= \frac{e^{\frac{\cos(2\pi x)+\cos(2\pi y)}{2}} \cdot 2\pi \cdot \frac{d}{dx}[x] \cdot \sin(2\pi x)}{2} + \frac{\sqrt{2}\, e^{-\frac{\sqrt{x^2+y^2}}{5\sqrt{2}}}\left(\frac{d}{dx}[x^2] + \frac{d}{dx}[y^2]\right)}{\sqrt{x^2+y^2}}$$

$$= \pi e^{\frac{\cos(2\pi x)+\cos(2\pi y)}{2}} \cdot 1 \sin(2\pi x) + \frac{\sqrt{2}^{-\frac{\sqrt{x^2+y^2}}{5\sqrt{2}}}\, 2x + 0}{\sqrt{x^2+y^2}}$$

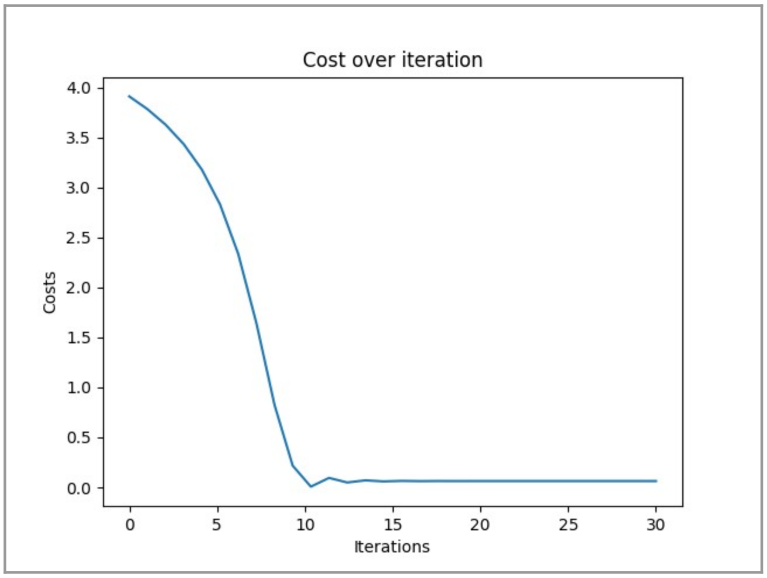
$$= \pi e^{\frac{\cos(2\pi x)+\cos(2\pi y)}{2}} \cdot \sin(2\pi x) + \frac{2^{\frac{3}{2}} x e^{-\frac{\sqrt{x^2+y^2}}{5\sqrt{2}}}}{\sqrt{x^2+y^2}}$$ ... for x gradient

$$\pi e^{\frac{\cos(2\pi y)+\cos(2\pi x)}{2}} \cdot \sin(2\pi y) + \frac{2^{\frac{3}{2}} y e^{-\frac{\sqrt{y^2+x^2}}{5\sqrt{2}}}}{\sqrt{y^2+x^2}}$$ ... for y gradient

3.4)

We chose following parameters: learning_rate = 0.01, lr_decay = 0.99, max_iter = 30.

3.5)



3.6)
By choosing a random variable between 0 and 1 for our starting point, our algorithm will have found a minimum very quickly. However, since this lies behind the starting point, it is not the minimum we are looking for. So we always have to find a suitable decay and learning rate to prevent these results.

3.7)
It is really important to have a learning rate that decays. Otherwise we maybe would not be able find the global minimum. We probably would overshoot the mark of the minimum. With the decay our learning rate is getting smaller. In this way we can approach the minimum with smaller and smaller steps in order to finally reach it.