

P1

February 17, 2021

0.1 P1

0.2 a)

```
[2]: from matplotlib import pyplot as plt
import numpy as np
```

```
[36]: def huber_loss(delta, y, t):
    a = y - t

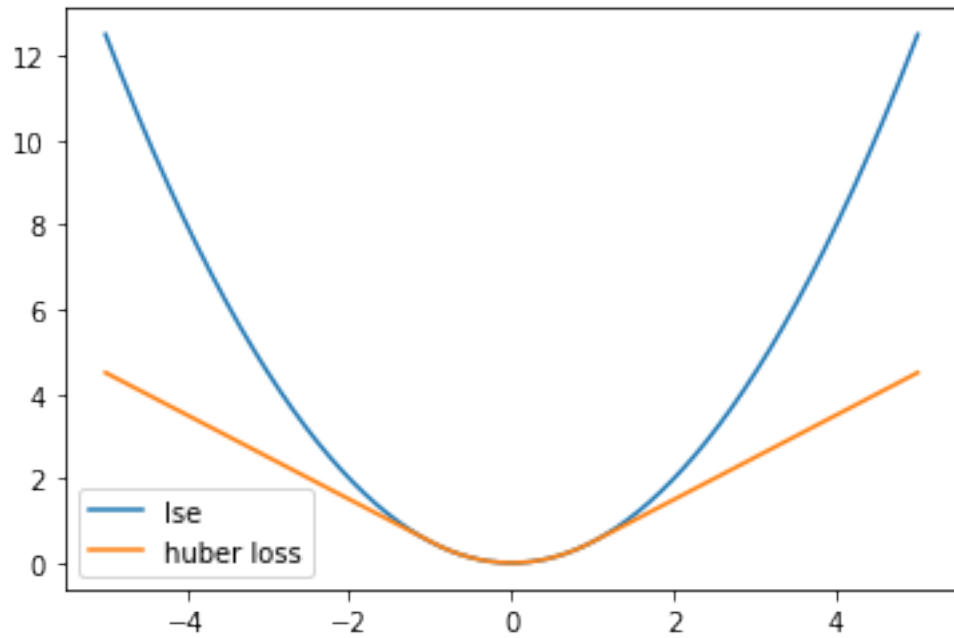
    mask_le = (abs(a) <= delta).astype(int)
    mask_g = 1 - mask_le

    return 0.5 * a ** 2 * mask_le + delta * (abs(a) - 0.5 * delta) * mask_g
```

```
[37]: lse = lambda y, t: 0.5 * (y - t) ** 2
```

```
[39]: y = np.linspace(-5, 5, 1001)
t = 0

plt.plot(y, lse(y, 0), label='lse')
plt.plot(y, huber_loss(1, y, t), label='huber loss')
plt.legend()
plt.show()
```



As we can see, huber loss does not penalize outliers as severely as least squared error (It acts linearly not quadratically in those regimes). Therefore it is less sensitive to outliers.