MALIS
Group Exercise

November 15 2022

Group Name:	
Group Members:	

1 Regression, Gradient Descent and the Perceptron

Derive a gradient descent training algorithm that minimizes the sum of squared errors for a variant of a perceptron where the output o depends on its units as follows:

$$o = w_0 + w_1 x_1 + w_1 x_1^3 + w_2 x_2 + w_2 x_2^3 + \ldots + w_d x_d + w_d x_d^3$$

Give your answer in the form $w_i \leftarrow w_i + \ldots$ for $1 \leq i \leq d$

G.D update rule $\omega^{(\tau+1)} \leftarrow \omega^{(\tau)} - \alpha \nabla_W \text{Loss}$

$\text{Loss} = \sum (y - \hat{o})^2$

ground truth

$\frac{\partial \text{Loss}}{\partial w_i} = -2 \sum^N (y - \hat{o}) \cdot (x_i + x_i^3)$

batch G.D. $w_i \longleftarrow w_i + \alpha \cdot 2 \sum_{j=1}^{N} (y_j - \hat{o}_j) \cdot (x_i + x_i^3)$

stochastic G.D. $w_i \longleftarrow w_i + 2\alpha (y - \hat{o})(x_i + x_i^3)$

2 Support Vector Machines

You train an SVM using N training points. You observe that the trained model has M support vectors. A new set of K points arrives. You retrain your SVM using N + K points. Can you tell how many support vectors your new model will have?

Not with the available information.

You would need to know where are the new K points located with respect to the M support vectors and the decision boundary