Mathematical Foundation of Machine Learning Spring 2020: Assignment III

1. (30%) Let $\mathcal{F}$ be a forest of trees. Assume there exists $0 < \rho < 1$, such that for any $\Omega \in \mathcal{F}$ on level l and any $\Omega' \in \mathcal{F}$ on level $l+1$, with $\Omega \bigcap \Omega' \neq \emptyset$, we have $|\Omega'| \leq \rho |\Omega|$. In class we proved the following: For any $\Omega' \in \mathcal{F}$ and any $x \in \Omega'$, let $\Lambda_x := \{\Omega \in \mathcal{F} : x \in \Omega,\ |\Omega| \geq |\Omega'|\}$. Then, for $0 < p < \infty$,

$$\sum_{\Omega \in \Lambda_x} \left(\frac{|\Omega'|}{|\Omega|} \right)^{1/p} \leq c(\rho, p) J .$$

   Now define $\Lambda_{\Omega'} := \{\Omega \in \mathcal{F} : \Omega \bigcap \Omega' \neq \emptyset,\ |\Omega| \geq |\Omega'|\}$. Prove or disprove that there exists a constant $c > 0$, such that for any $\Omega' \in \mathcal{F}$,

$$\sum_{\Omega \in \Lambda_{\Omega'}} \left(\frac{|\Omega'|}{|\Omega|} \right)^{1/p} \leq c .$$

2. (30%) [Besov smoothness of indicator functions] Let $f(x) = 1_{\tilde{\Omega}}(x)$, where $\tilde{\Omega} \subset [0,1]^2$ is a domain with a smooth boundary. Assume that one can construct an anisotropic tree $\mathcal{T}_A$, such that from some level $k_0 \geq 1$, there are only $\leq c_1 2^m$ domains at the level $k_0 + 2m$ that intersect the boundary of $\tilde{\Omega}$, each of area $\leq c_2 2^{-3m}$. Prove that $f \in B_\tau^\alpha(\mathcal{T}_A)$, where $\alpha < 2/3\tau$. You can assume that there exists $0 < \rho < 1$, such that for any child Ω' of Ω, $|\Omega'| \leq \rho |\Omega|$.

3. (30%) [Back propagation] Let $f(x) = h_\theta\left(f_2\left(f_1(x, w_1), w_2\right)\right)$, $w := (w_1, w_2) \in \mathbb{R}^2$, $\theta = (\beta, \beta_0) \in \mathbb{R}^2$, be a binary classification model consisting of 2 neural network layers and a logistic regression function

$$h_\theta(t) = \frac{1}{1 + e^{-(\beta t + \beta_0)}} .$$

   Compute the gradient with respect to the negative log-likelihood loss, for the 4 weights at some point $(w^*, \theta^*) \in \mathbb{R}^4$, using the training dataset $\{x_i, y_i\}_{i \in I}$, $x_i \in \mathbb{R}$, $y_i \in \{0,1\}$.

4. (10%) The random forest model is stronger learner than the linear regression model. Can one build a regression model composed of a neural network with a random forest at the last layer instead of a linear regression model?