Homework 3 - Version 1.1

Deadline: Thurs, Mar.5, at 11:59pm.

Submission: You must submit your solutions as a PDF file through MarkUs[1]. You can produce the file however you like (e.g. LaTeX, Microsoft Word, scanner), as long as it is readable.

See the syllabus on the course website[2] for detailed policies. You may ask questions about the assignment on Piazza[3]. *Note that 10% of the homework mark (worth 1 pt) may be removed for a lack of neatness.*

The teaching assistants for this assignment are Cem Anil and Sheng Jia.
`mailto:csc413-2020-01-tas@cs.toronto.edu`

1 Weight Decay

Here, we will develop further intuitions on how adding weight decay can influence the solution space. For a refresher on generalization, please refer to: `https://csc413-2020.github.io/assets/readings/L07.pdf`. Consider the following linear regression model with weight decay.

$$\mathcal{J}(\hat{\mathbf{w}}) = \frac{1}{2n}\|X\hat{\mathbf{w}} - \mathbf{t}\|_2 + \frac{\lambda}{2}\hat{\mathbf{w}}^\top\hat{\mathbf{w}}$$

where $X \in \mathbb{R}^{n\times d}$, $Y \in \mathbb{R}^n$, and $\hat{\mathbf{w}} \in \mathbb{R}^d$. n is the number of data points and d is the data dimension. X is the design matrix in HW1.

1.1 Underparameterized Model [0pt]

First consider the underparameterized $d \leq n$ case. Write down the solution obtained by gradient descent assuming training converges. Is the solution unique? If the solution involves inverting matrices, explain why it is invertible.

1.2 Overparameterized Model

1.2.1 Warmup: Visualizing Weight Decay [1pt]

Now consider the overparameterized $d > n$ case. We start with a 2D example from HW1. For a single training example $\mathbf{x}_1 = [2, 1]$ and $t_1 = 2$. First, 1) draw the solution space of the squared error on a 2D plane. Then, 2) draw the the coutour plot of the weight decay term $\frac{\lambda}{2}\hat{\mathbf{w}}^\top\hat{\mathbf{w}}$.

Include the plot in the report. Also indicate on the plot where the gradient descent solutions are with and without weight decay. (Precious drawings are not required for the full mark.)

[1] `https://markus.teach.cs.toronto.edu/csc413-2020-01`
[2] `https://csc413-2020.github.io/assets/misc/syllabus.pdf`
[3] `https://piazza.com/class/k58ktbdnt0h1wx?cid=1`

1.2.2 Gradient Descent and Weight Decay [0pt]

Derive the solution obtained by gradient descent at convergence in the overparameterized case. Is this the same solution from Homework1 3.4.1?

1.3 Adaptive optimizer and Weight Decay [1pt]

In HW2 Section 1.2, we saw that per-parameter adaptive methods, such as AdaGrad, Adam, do not converge to the least norm solution due to moving out of the row space of our design matrix X.

Assume AdaGrad converges to an optimal in the training objective. Does weight decay help AdaGrad converge to a solution in the row space? Give a brief justification.

(Hint: build intuition from the 2-D toy example.)

2 Ensembles and Bias-variance Decomposition

In the prerequisite CSC311 `https://amfarahmand.github.io/csc311/lectures/lec04.pdf`, we have seen the bias-variance decomposition. The following question uses the same notation as taught in CSC311. As a reminder, the expected generalization error is the following:

$$\underbrace{\mathbb{E}\left[|h(x;\mathcal{D}) - y_*(x)|^2\right]}_{\text{① generalization err}} = \underbrace{\mathbb{E}\left[|\mathbb{E}[h(x;\mathcal{D})\,|\,x] - y_*(x)|^2\right]}_{\text{② bias}} + \underbrace{\mathbb{E}\left[|h(x;\mathcal{D}) - \mathbb{E}[h(x;\mathcal{D})\,|\,x]|^2\right]}_{\text{③ variance}} + \underbrace{\mathbb{E}\left[|y_*(x) - y|^2\right]}_{\text{Irreducible error}}$$

where x, y represent the sampled test data. $\mathcal{D}$ represents the sampled training dataset. $h(\cdot;\mathcal{D})$ is our prediction model learned using this dataset (i.e. $h(\cdot;\mathcal{D})$ is the learnt hypothesis given the training examples). $y_*(x)$ is the true model we want to learn. In the following questions, we are interested in the generalization performance of ensemble.

2.1 Weight Average or Prediction Average?

2.1.1 [1pt]

Does the ensemble of linear models using weight average or prediction average give the same expected generalization error? Provide a mathematical justification.

2.1.2 [0pt]

Does the ensemble of (nonlinear) neural networks using weight average or prediction average give the same expected generalization error? Provide a mathematical justification.

2.2 Bagging - Uncorrelated Models

One way to construct an ensemble is through bootstrap aggregation, or bagging, that takes a dataset $\mathcal{D}$ and generates k new datasets, with replacement. In this question, we assume the generated dataset $\mathcal{D}_i$ has the *same* size as $\mathcal{D}$. Then train a model for each dataset, $\mathcal{D}_i$, resulting in k models. The ensemble model is following:

$$\bar{h}(x;\mathcal{D}) = \frac{1}{k}\sum_{i=1}^{k} h(x;\mathcal{D}_i)$$

For this part, we will make a very unrealistic assumption that the predictions of the ensemble members are uncorrelated. That is $\mathrm{Cov}(h(x;\mathcal{D}_j), h(x;\mathcal{D}_k)) = 0$.

2.2.1 Bias with bagging [0pt]

Show that ensemble does not change the bias term in the generalization error.

$$\text{Show } bias = \mathbb{E}\left[\left|\mathbb{E}\left[\bar{h}(x;\mathcal{D}) \mid x\right] - y_*(x)\right|^2\right] = \mathbb{E}\left[\left|\mathbb{E}[h(x;\mathcal{D}) \mid x] - y_*(x)\right|^2\right]$$

2.2.2 Variance with bagging [1pt]

Assume the variance of a single predictor is σ^2, $\mathbb{E}\left[\left|h(x;\mathcal{D}) - \mathbb{E}[h(x;\mathcal{D}) \mid x]\right|^2\right] = \sigma^2$. Derive the variance term of ensemble in terms of σ under uncorrelated predictors.

2.3 Bagging - General Case

In practice, there will be correlations among the k predictions of the ensemble members because the sampled training datasets would be very similar to each other. For simplicity, assume a non-zero pairwise correlation between the ensemble members, that is ρ. The variance of the predictor for $h(x;\mathcal{D}_j)$ is σ_j^2.

$$\rho = \frac{\mathrm{Cov}(h(x;\mathcal{D}_j), h(x;\mathcal{D}_k))}{\sigma_j \sigma_k} \qquad \forall j \neq k$$

2.3.1 Bias under Correlation [1pt]

Does the correlation change the bias term in the generalization error? If so, derive the new expression in terms of ρ. Provide your justification.

2.3.2 Variance under Correlation [0pt]

Assume the variance of a single predictor is σ^2. Derive the variance term of ensemble in terms of σ and ρ.

$$\text{Show } variance = \mathbb{E}\left[\left|\bar{h}(x;\mathcal{D}) - \mathbb{E}\left[\bar{h}(x;\mathcal{D})|x\right]\right|^2\right] = \left(\rho + \frac{1-\rho}{k}\right)\sigma^2$$

2.3.3 Intuitions on bagging [1pt]

Based on the derived variance, what happens to the variance when you increase the number of ensemble models k? What do $\rho = 0$, $\rho = 1$ represent and their consequences for the variance?

3 Generalization and Dropout

In this question, we will investigate the effect of using dropout on generalization.
Lets say we generate triples (X_1, X_2, Y) according to the following model:

$$X_1 \leftarrow Gaussian(0, \sigma^2) \tag{3.1}$$

$$Y \leftarrow X_1 + Gaussian(0, \sigma^2) \tag{3.2}$$

$$X_2 \leftarrow Y + Gaussian(0, 1) \tag{3.3}$$

3.1 Regression Coefficients

We will try to predict Y from (X_1, X_2) using a linear least-square predictor. To be specific, we'll try to minimize $\mathcal{J} = \frac{1}{2N}\sum_1^N (y^{(i)} - \hat{y}^{(i)})^2$ where $\hat{y} = w_1 x_1 + w_2 x_2$ and N is the size of the samples in our training set. In the remainder of the question, you can assume that we have infinitely many samples - in this case, we can write the loss as $\mathcal{J} = \mathbb{E}_{(x_1,x_2,y)\sim(X_1,X_2,Y)}[(y^{(i)} - \hat{y}^{(i)})^2]$

3.1.1 Regress from X_1 [0pt]

Find the value of w_1 if we're only using X_1 to predict Y. Your answer might depend on σ.
Hints: You may consider taking the following steps:

- Step 1: Write out the error as an expectation under the random variable X_1, X_2 and Y.
- Step 2: Enter the equations from the structural equation model into the error expression.
- Step 3: Take the square, and separate the expectations. Many terms cancel due to independence.
- Step 4: Differentiate the final expression wrt. w_1 and w_2 and set it to 0. Solve for the optimal values of alpha.

3.1.2 Regress from X_2 [1pt]

Find the value of w_2 if we're only using X_2 to predict Y. Your answer might depend on σ.

3.1.3 Regress from (X_1, X_2) [1pt]

1) Find (w_1, w_2) if we're using (X_1, X_2) to predict Y in terms of σ. 2) Comment on if this maximum likelihood solution will generalize well during the test time if σ changes.

(Hint: think about the causal relationship of the random variables. What happens if σ becomes smaller during the test time)

3.1.4 Different σs. [0pt]

Now assume half of the dataset was sampled using $\sigma = \sigma_1$ and the other half was sampled using $\sigma = \sigma_2$. How does the coefficients computed in 3.1.3 change?

3.2 Dropout as Data-Dependent $L2$ Regularization

Lets say we now apply dropout to X_1 and X_2. That is, we now have $\hat{y} = 2(m_1 w_1 x_1 + m_2 w_2 x_2)$ where both m_1 and m_2 are iid. Bernoulli variables with expectation $\frac{1}{2}$. Note that since the dropout mask is a random variable, the output of our model is also a random variable. You may find Slide 21 in Lecture 7 `https://csc413-2020.github.io/assets/readings/L07.pdf` helpful in answering this question. 3.2.1 is no-credit review question whose answers can be found directly in slides.

3.2.1 Expectation and variance of predictions [0pt]

Show that $\mathbb{E}[\hat{y}] = w_1 x_1 + w_2 x_2$ and $Var[\hat{y}] = w_1^2 x_1^2 + w^2 x_2^2$.

3.3 Effect on Dropout [1pt]

If we're using both (X_1, X_2) to predict Y, how does applying dropout change the optimal parameters? Will this solution generalize better than 3.1.3? Why?

(Hint: try directly minimizing the expected loss under dropout, the last equation on Slide 21 `https://csc413-2020.github.io/assets/readings/L07.pdf`.)