Homework 8 - Solution

1. (a) Let $s < 1/\beta$. We have by Proposition 2.8 (characterization of β-smoothness) that

$$\begin{aligned} f(x^t - s\nabla f(x^t)) &\leq f(x^t) - \langle \nabla f(x^t), s\nabla f(x^t)\rangle + \frac{\beta}{2}\|s\nabla f(x^t)\|^2 \\ &= f(x^t) - \|\nabla f(x^t)\|^2 \left(s - \frac{\beta}{2}s^2\right) \\ &= f(x^t) - s\|\nabla f(x^t)\|^2 \left(1 - \frac{\beta}{2}s\right) \\ &\leq f(x^t) - \frac{s}{2}\|\nabla f(x^t)\|^2 < f(x^t) - \lambda s\|\nabla f(x^t)\|^2 \end{aligned}$$

as $\lambda < 1/2$. Therefore, if $s < 1/\beta$, the backtracking line search stops. After L loops, starting at $s = 1$, we have $s = \mu^L$. So the maximal number of iterations is $L = \max(1, \log(\beta)/\log(1/\mu))$.

(b) If $1 < 1/\beta$, we stop before making any iterations of the loop so that the final value of s is 1. Assume now that $1 \geq 1/\beta$. Just before the last iteration, we have $s \geq 1/\beta$ (otherwise we would stop here). So, at the next iteration, when we stop, the value of s is equal to μ times the previous value of s, which is therefore larger than μ/β. Putting both cases together, the backtracking line search terminates with a value $s \geq \min(1, \mu/\beta)$. Plugging this information in the stopping condition gives

$$f(x^{t+1}) \leq f(x^t) - \lambda \min(1, \mu/\beta)\|\nabla f(x^t)\|^2.$$

(c) According to the PL inequality, we have

$$\|\nabla f(x^t)\|^2 \geq 2\alpha(f(x^t) - f(x^\star)).$$

Therefore,

$$\begin{aligned} f(x^{t+1}) - f(x^\star) &\leq f(x^t) - f(x^\star) - \lambda \min(1, \mu/\beta) 2\alpha (f(x^t) - f(x^\star)) \\ &= (f(x^t) - f(x^\star))(1 - 2\alpha\lambda \min(1, \mu/\beta)). \end{aligned}$$

By iterating this equation, we obtain the result. Several remarks are to be made. First, the rate of convergence of this algorithm is still linear. However, the precise rate of convergence is slower than for gradient descent with constant step size as $2\alpha\lambda \min(1, \mu/\beta) \leq \alpha/\beta$ (where α/β was the constant appearing for the vanilla gradient descent). Still, backtracking line search has a huge advantage over the constant step-size algorithm: it can be implemented without knowing the parameter β of smoothness of f. Also, it is not much more costly to implement than constant step size gradient descent (only $\max(1, \log(\beta)/\log(1/\mu)$ computations are needed at each iteration of the gradient descent).

2. (a) In the proof of Proposition 4.2, at iteration t, only the behavior of f on the line $[x^\star, x^t]$ is of interest. Therefore, if the restriction of f on this $[x^\star, x^t]$ is α-strongly convex, β-smooth and has a Hessian that is γ-Lipschitz continuous, we can conclude as in the proof of Proposition 4.2 (see Eq. (27)) that $\|x^{t+1} - x^\star\| \leq \frac{\gamma}{2\alpha}\|x^t - x^\star\|^2$. If $\|x^t - x^\star\|^2 \leq 2\alpha/\gamma$, this implies that $\|x^{t+1} - x^\star\| \leq \|x^t - x^\star\|$. If $\|x^0 - x^\star| \leq 2\alpha/\gamma$ and f satisfies those assumptions on the ball $B(x^\star, \|x^0 - x^\star\|)$, we can therefore prove by induction that the restriction of f on the line $[x^\star, x^t]$ always satisfies the required properties, allowing us to obtain the same convergence result.

   (b) By symmetry, the minimizer $x^\star$ is 0. Computations yield

$$\begin{aligned} f'(x) &= 1 - \frac{2}{1 + e^{2x}} \\ f''(x) &= \frac{4e^{2x}}{(1 + e^{2x})^2} \end{aligned}$$

Note that $f'' \geq 0$ so that f is indeed convex. The function f'' is even and decreasing on $[0, +\infty)$. Its minimal value is 0 whereas its maximal value is 1. Therefore, it is 1-smooth ($\beta = 1$) and α-strongly convex only for $\alpha = 0$ (that is it is **not** strongly convex).

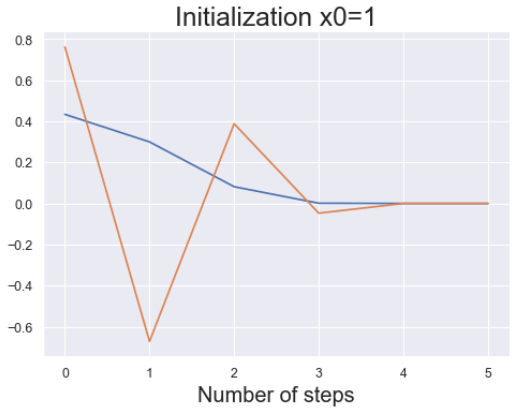

Figure 1: Initialization at $x_0 = 1$.

(c) Consider the initialization $x^0 = 1$. According to question 2., the restriction of f on $[-1, 1]$ is α-strongly convex with $\alpha = f''(1)$. We have

$$\frac{2\alpha}{\gamma} = \frac{3\sqrt{3} \cdot 4e^2}{2(1+e^2)^2} \approx 1.092 > 1 = \|x^0 - x^\star\|.$$

Therefore, according to question 1., Newton's method should converge. We observe indeed a very fast convergence in Figure 1.

(d) We have, with $\alpha = f''(1.1)$,

$$\frac{2\alpha}{\gamma} = \frac{3\sqrt{3} \cdot 4e^{2.2}}{2(1+e^{2.2})^2} \approx 0.933 < 1.1 = \|x^0 - x^\star\|.$$

Therefore, question 1. does not imply that Newton's method should converge (it does not also imply that is should diverge). We observe that Newton's method quickly diverges in Figure 2. This showcases the fact that Newton's method only converges on a small neighborhood of the minimum: initialization has to be chosen carefully!

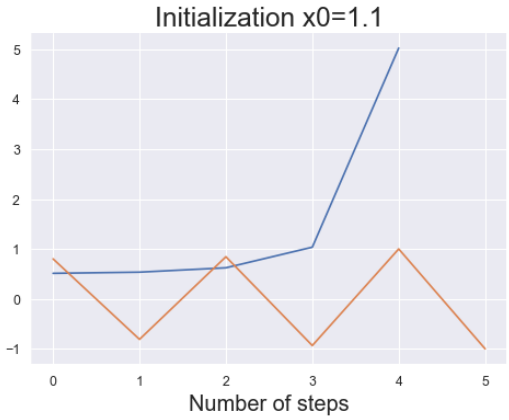

Figure 2: Initialization at $x_0 = 1.1$.

3. (a) We have (recalling that $f_j(\mathbf{x_i}) \in \{-1, +1\}$)

$$\begin{aligned}
\mathcal{R}_n(\hat{F}_{t-1}+\alpha f_j) &= \frac{1}{n}\sum_{i=1}^{n} \exp(-\mathbf{y_i}(\hat{F}_{t-1}(\mathbf{x_i}) + \alpha f_j(\mathbf{x_i})) \\
&= \sum_{i=1}^{n} w_i^{(t)} \exp(-\alpha \mathbf{y_i} f_j(\mathbf{x_i})) \\
&= \sum_{i=1}^{n} w_i^{(t)} (e^{-\alpha}\mathbf{1}\{\mathbf{y_i} = f_j(\mathbf{x_i})\} + e^{\alpha}\mathbf{1}\{\mathbf{y_i} \neq f_j(\mathbf{x_i})\}) \\
&= e^{-\alpha}\sum_{i=1}^{n} w_i^{(t)} (1 - \mathbf{1}\{\mathbf{y_i} \neq f_j(\mathbf{x_i})\} + e^{2\alpha}\mathbf{1}\{\mathbf{y_i} \neq f_j(\mathbf{x_i})\}) \\
&= e^{-\alpha}\sum_{i=1}^{n} w_i^{(t)} (1 + (e^{2\alpha} - 1)\mathbf{1}\{\mathbf{y_i} \neq f_j(\mathbf{x_i})\}) \\
&= e^{-\alpha}\sum_{i=1}^{n} w_i^{(t)} (1 + (e^{2\alpha} - 1)\varepsilon_t(j))
\end{aligned}$$

Let us fix j. Then, this function is a strictly convex function in

α, whose derivative is given by

$$\begin{aligned}
&e^{-\alpha}\sum_{i=1}^{n} w_i^{(t)}\left(-(1+(e^{2\alpha}-1)\varepsilon_t(j))+2e^{2\alpha}\varepsilon_t(j)\right)\\
&=e^{-\alpha}\sum_{i=1}^{n} w_i^{(t)}\left(-1+\varepsilon_t(j)+e^{2\alpha}\varepsilon_t(j)\right)\\
&=e^{-\alpha}\sum_{i=1}^{n} w_i^{(t)}\left(\varepsilon_t(j)(1+e^{2\alpha})-1\right).
\end{aligned}$$

Therefore, the minimum of $\alpha \mapsto \mathcal{R}_n(\hat{F}_{t-1}+\alpha f_j)$ is attained at $\alpha_j = \frac{1}{2}\log\left(\frac{1-\varepsilon_t(j)}{\varepsilon_t(j)}\right)$. The optimal value of j is the one that minimizes

$$\begin{aligned}
&\mathcal{R}_n(\hat{F}_{t-1}+\alpha_j f_j)\\
&=e^{-\alpha_j}\sum_{i=1}^{n} w_i^{(t)}(1+(e^{2\alpha_j}-1)\varepsilon_t(j))\\
&=\sqrt{\frac{\varepsilon_t(j)}{1-\varepsilon_t(j)}}\sum_{i=1}^{n} w_i^{(t)}\left(1+\left(\frac{1-\varepsilon_t(j)}{\varepsilon_t(j)}-1\right)\varepsilon_t(j)\right)\\
&=\sqrt{\frac{\varepsilon_t(j)}{1-\varepsilon_t(j)}}\sum_{i=1}^{n} w_i^{(t)}\left(2-2\varepsilon_t(j)\right)\\
&=2\sum_{i=1}^{n} w_i^{(t)}\sqrt{\varepsilon_t(j)(1-\varepsilon_t(j))}.
\end{aligned}$$

As each $\varepsilon_t(j)$ is smaller than $1/2$, the minimum is attained for the smallest $\varepsilon_t(j)$, that we call E_t.

(b) The complexity of computing T steps is $O(Tm)$.

(c) Note that $\sum_{i=1}^{n} w_i^{(t)} = \mathcal{R}_n(\hat{F}_{t-1})$ by definition. Therefore, according to the previous question,

$$\mathcal{R}_n(\hat{F}_t) = 2\mathcal{R}_n(\hat{F}_{t-1})\sqrt{E_t(1-E_t)}.$$

We conclude by iterating this equation, and by noting that $\mathcal{R}_n(\hat{F}_0) = 1$ as $\hat{F}_0 = 0$.

(d) As the function $x \mapsto 4x(1-x)$ is increasing on $[0, 1/2]$, we have

$$\begin{aligned} 4E_t(1-E_t) &\leq 4(1/2-\gamma)(1/2+\gamma) = 4(1/4-\gamma^2) \\ &= 1 - 4\gamma^2 \leq \exp(-4\gamma^2), \end{aligned}$$

where we use the inequality $1 - t \leq \exp(-t)$. This inequality and the previous question allow us to conclude.

(e) It holds that

$$\begin{aligned} \mathbb{E}[\tilde{\mathcal{R}}_P(\hat{G})] &= \mathbb{E}[\tilde{\mathcal{R}}_n(\hat{G})] + \mathbb{E}[\tilde{\mathcal{R}}_P(\hat{G}) - \tilde{\mathcal{R}}_n(\hat{G})] \\ &\leq \mathbb{E}[\tilde{\mathcal{R}}_n(\hat{G})] + \mathbb{E}[\sup_{G \in \mathcal{G}} (\tilde{\mathcal{R}}_P(G) - \tilde{\mathcal{R}}_n(G))]. \end{aligned}$$

The first quantity is smaller than $\exp(-2T\gamma^2)$ according to the previous question, whereas the second quantity is bounded by the second term in the inequality appearing in the question according to Theorem 3.7 in Chapter 1.

(f) We plot in Figure 3 both the empirical risk for the $0-1$ loss on the training sample and on the testing sample. For $T = 1$, we are using the best stump classifier, that is as expected not very accurate (20% of errors) on both the training set and the testing set. This makes sense because this classifier is too simple: the approximation error is large. When T gets larger, as expected, the empirical risk decreases very quickly to 0. This is a consequence of question (d). What is surprising is that, even for T very large, we do not see the testing error increases. Our current understanding of the problem would indicate that, for T very large, we should be overfitting and the testing error should get larger (this is reflected by the second term in question (e) getting larger with T). This is not reflected in this experiment.

It turns out that a more thorough study of those predictors can explain why overfitting does not happen when using the AdaBoost algorithm in some experiments. The student interested by the theory can check the book "Boosting - Foundations and Algorithms" by Schapire and Freud (Chapter 5).

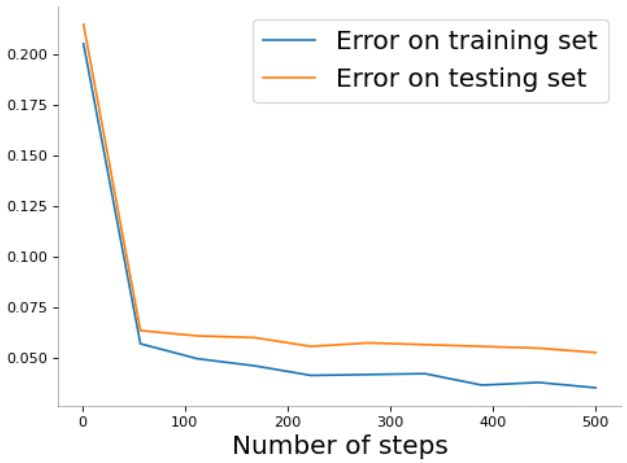

Figure 3: Training and testing error on the spam set for different number of iterations of the AdaBoost algorithm.