Fast Approximate Regression

Lecture 21

Nov 10, 2022

Linear least squares/Regression

Linear least squares: Given $\boldsymbol{A} \in \mathbb{R}^{\boldsymbol{n} \times \boldsymbol{d}}$ and $\boldsymbol{b} \in \mathbb{R}^{\boldsymbol{d}}$ find $\boldsymbol{x}$ to minimize $\|\boldsymbol{Ax} - \boldsymbol{b}\|_2$.

Interesting when $\boldsymbol{n} \gg \boldsymbol{d}$ the over constrained case when there is no solution to $\boldsymbol{Ax} = \boldsymbol{b}$ and want to find best fit.

Geometrically $\boldsymbol{Ax}$ is a linear combination of columns of $\boldsymbol{A}$. Hence we are asking what is the vector $\boldsymbol{z}$ in the column space of $\boldsymbol{A}$ that is closest to vector $\boldsymbol{b}$ in ℓ_2 norm.

Closest vector to $\boldsymbol{b}$ is the projection of $\boldsymbol{b}$ into the column space of $\boldsymbol{A}$ so it is "obvious" geometrically. How do we find it?

Linear least squares/Regression

Linear least squares: Given $A \in \mathbb{R}^{n \times d}$ and $b \in \mathbb{R}^d$ find x to minimize $\|Ax - b\|_2$.

Interesting when $n \gg d$ the over constrained case when there is no solution to $Ax = b$ and want to find best fit.

Geometrically Ax is a linear combination of columns of A. Hence we are asking what is the vector z in the column space of A that is closest to vector b in ℓ_2 norm.

Closest vector to b is the projection of b into the column space of A so it is "obvious" geometrically. How do we find it? Find an orthonormal basis $z_1, z_2, \ldots, z_r$ for the columns of A. Compute projection c as $c = \sum_{j=1}^{r} \langle b, z_j \rangle z_j$ and output answer as $\|b - c\|_2$.

Linear least square/Regression and SVD

Linear least squares: Given $\boldsymbol{A} \in \mathbb{R}^{\boldsymbol{m} \times \boldsymbol{n}}$ and $\boldsymbol{b} \in \mathbb{R}^{\boldsymbol{m}}$ find $\boldsymbol{x}$ to minimize $\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|_2$.

Closest vector to $\boldsymbol{b}$ is the projection of $\boldsymbol{b}$ into the column space of $\boldsymbol{A}$ so it is "obvious" geometrically. Find an orthonormal basis $z_1, z_2, \ldots, z_r$ for the columns of $\boldsymbol{A}$. Compute projection $\boldsymbol{b}'$ as $\boldsymbol{b}' = \sum_{j=1}^{r} \langle \boldsymbol{b}, z_j \rangle z_j$ and output answer as $\|\boldsymbol{b} - \boldsymbol{b}'\|_2$.

Finding the basis is the expensive part. Recall SVD gives $v_1, v_2, \ldots, v_r$ which form a basis for the *row* space of $\boldsymbol{A}$ but then $u_1^T, u_2^T, \ldots, u_m^T$ form a basis for the *column* space of $\boldsymbol{A}$. Hence SVD gives us all the information to find $\boldsymbol{b}'$. In fact we have

$$\min_{\boldsymbol{x}} \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|_2^2 = \sum_{i=r+1}^{m} \langle u_i^T, \boldsymbol{b} \rangle^2$$

Subspace Embedding

Question: Suppose we have linear subspace E of $\mathbb{R}^n$ of dimension d. Can we find a projection $\Pi : \mathbb{R}^n \rightarrow \mathbb{R}^k$ such that for *every* $x \in E$, $\|\Pi x\|_2 = (1 \pm \epsilon)\|x\|_2$?

- Not possible if $k < d$.
- Possible if $k = d$. Pick Π to be an orthonormal basis for E. **Disadvantage:** This requires knowing E and computing orthonormal basis which is slow.

What we really want: *Oblivious* subspace embedding ala JL based on random projections

Oblivious Supspace Embedding

Theorem

Suppose E is a linear subspace of $\mathbb{R}^n$ of dimension d. Let Π be a DJL matrix $\Pi \in \mathbb{R}^{k \times n}$ with $k = O(\frac{d}{\epsilon^2} \log(1/\delta))$ rows. Then with probability $(1-\delta)$ for every $x \in E$,

$$\|\frac{1}{\sqrt{k}}\Pi x\|_2 = (1 \pm \epsilon)\|x\|_2.$$

In other words JL Lemma extends from one dimension to arbitrary number of dimensions in a graceful way.

Linear least squares via Subspace embeddings

Let $a_1, a_2, \ldots, a_d$ be the columns of A and let E be the subspace spanned by $\{a_1, a_2, \ldots, a_d, b\}$

E has dimension at most $d + 1$.

Use subspace embedding on E. Applying JL matrix Π with $k = O(\frac{d}{\epsilon^2})$ rows we reduce $a_1, a_2, \ldots, a_d, b$ to $a'_1, a'_2, \ldots, a'_d, b'$ which are vectors in $\mathbb{R}^k$.

Solve $\min_{x' \in \mathbb{R}^d} \|A'x' - b'\|_2$

Faster Linear least squares via Subspace embeddings

Let $a_1, a_2, \ldots, a_d$ be the columns of A and let E be the subspace spanned by $\{a_1, a_2, \ldots, a_d, b\}$

E has dimension at most $d+1$.

Use subspace embedding on E. Applying JL matrix Π with $k = O(\frac{d}{\epsilon^2})$ rows we reduce $a_1, a_2, \ldots, a_d, b$ to $a'_1, a'_2, \ldots, a'_d, b'$ which are vectors in $\mathbb{R}^k$.

Solve $\min_{x' \in \mathbb{R}^d} \|A'x' - b'\|_2$

Claim: Answer is a $(1+O(\epsilon))$-approximation to original problem if Π is a $(1+\epsilon)$-approximate subspace embedding.

Faster Linear least squares via Subspace embeddings

Apply subspace embedding Π to A, b to obtain A', b'

Solve $\min_{x' \in \mathbb{R}^d} \|A'x' - b'\|_2$

Claim: Answer is a $(1 + O(\epsilon))$-approximation to original problem if Π is a $(1 + \epsilon)$-approximate subspace embedding.

Advantage: Reduces A from $n \times d$ to $k \times d$ where $k = O(d/\epsilon^2)$. Use any fast approximate regression method on A', b' as a black box.

Disadvantage: Dependence of $1/\epsilon^2$ is high if one wants to choose small ϵ. In particular if n and d are large and comparable.

Accelerating Iterative Solvers via Sketching

- Iterative solvers that converge to solution are very common in numerical linear algebra. Each iteration is fast and goal is to reduce number of iterations
- Typically the number of iterations depends on how well-behaved the data is. An example is the *condition number* of the matrix.
- Iterative solvers can be sped up by *pre-conditioning* to make data well-behaved.

Goal: show that sketching techniques such as oblivious supspace embeddings can be viewed as preconditioning tools. Demonstrate on least squares regression.

Gradient Descent

Let $f : \mathbb{R}^d \to \mathbb{R}$ be a real-valued differentiable function. Recall $\nabla f(x)$ is the gradient of f at x which is a vector in $\mathbb{R}^d$ with $(\nabla f(x))_i = \frac{\partial f}{\partial x_i}$. Gradient descent is a common search technique to find a local minimum/optimum of f in the unconstrained setting. A local optimum is a point x where $\nabla f(x) = 0$. When f is a convex function then any local optimum is a global optimum. There are many variants of gradient descent. Simplest one is based on having only access to the gradient and works with a fixed step size η.

GradientDescent(f, η)**:**
    `Choose a good strating point` $x^{(0)} \in \mathbb{R}$
    `For` $t = 1$ `to` T `to`
        $x^{(t)} \leftarrow x^{(t-1)} - \eta \nabla f(x^{(t-1)})$

    `Output` $x^{(T)}$

Gradient Descent

The choice of η (step size) is important for convergence and it depends on the smoothness of the function. If the gradient changes very rapidly it is difficult to find a local minimum since we may overshoot. An important parameter in the analysis is the smoothness which upper bounds the rate of change of the gradient.

Definition

f is L-smooth if $\|\nabla f(x) - \nabla f(y)\|_2 \leq L\|x - y\|_2$ for all x, y.

One can show that GD converges if $\eta \leq 1/L$. Convergence is much faster if the function is in addition *strongly* convex.

Convex functions

Definition

A real-valued continuous function $f : \mathbb{R}^d \rightarrow \mathbb{R}$ is convex over a domain $D \subseteq \mathbb{R}^d$ if for all $x, y \in D$ and for all $\theta \in [0,1]$, $f(\theta x + (1-\theta)y) \leq \theta f(x) + (1-\theta)f(y)$.

We will be interested in differentiable functions and twice-differentiable functions.

Fact: Differentiable function f is convex iff $f(x) \geq f(x_0) + (x - x_0)^T \nabla f(x_0)$ for all $x, x_0 \in D$.

f at any point x_0 lies above the tangent at point x_0.

f is *strictly* convex if $f(x) > f(x_0) + (x - x_0)^T \nabla f(x_0)$ for all x, x_0.

Convex functions

Suppose f is twice differentiable function. $H(x) = \nabla^2 f(x)$ is the Hessian of f at x. It is a $d \times d$ *symmetric* matrix where $H(x)_{i,j} = H(x)_{j,i} = \frac{\partial^2 f}{\partial x_i \partial x_j}$.

Fact: Twice-differentiable function f is convex iff $\nabla^2 f(x) \succeq 0$, that is, it is a positive semi-definite matrix. Alternatively, $y^T(\nabla^2 f(x))y \geq 0$ for all y, x.

A real-symmetric matrix has all real eigen values and hence $H(x)$ has real eigen-values for all twice-differentiable functions. When $H(x) \succeq 0$ (psd matrix) all the eigen-values are non-negative which means that the function's curvature is non-negative in all directions and hence bowl shaped (convex).

Strongly convex functions

Definition

A differentiable function f is *strongly convex* with parameter μ if $f(x) \geq f(x_0) + (\nabla f(x) - \nabla f(x_0))^T(x - x_0) + \frac{\mu}{2}\|x - x_0\|_2^2$ for all $x, x_0 \in D$. Equivalently, $(\nabla f(x) - \nabla f(x_0))^T(x - x_0) \geq \mu\|x - x_0\|_2^2$.

Fact: Twice differentiable f is strongly convex with parameter μ iff $\lambda_{\min}(H(x)) \geq \mu$ for all x where $\lambda_{\min}(H(x))$ is the smallest eigen-value of $H(x)$.

Fact: f is strongly convex with parameter μ iff the function $g(x) = f(x) - \frac{\mu}{2}\|x\|_2^2$ is convex.

Regression as convex optimization problem

Consider

$$f(x) = \|Ax - b\|_2^2 = x^T A^T A x - 2x^T A b + \|b\|_2^2$$

The gradient is easy to compute explicitly:

$$\nabla f(x) = 2A^T A x - 2Ab$$

One can see that the Hessian $\nabla^2 f(x) = 2A^T A$ and since $A^T A$ is psd it also shows that f is convex

Setting gradient to 0 one can see that the optimum solution value is $x^* = (A^T A)^{-1} A b$. Even though we have an explicit solution, iterative methods are preferred since maxtrix multiplication and computing the inverse are expensive.

Smoothness of regression

Suppose $f(x) = \|Ax - b\|_2^2 = x^T A^T Ax - 2x^T Ab + \|b\|_2^2$. It is a convex function with gradient $\nabla f(x) = 2A^T Ax - 2Ab$.

For x, y we have $\|\nabla f(x) - \nabla f(y)\|_2 = 2\|A^T A(x - y)\|_2$. It follows that

$$\|\nabla f(x) - \nabla f(y)\|_2 \leq 2\sigma_1^2 \|x - y\|_2$$

where σ_1 is the top singular value of A.

Thus, for the regression problem, f is L-smooth where $L = 2\sigma_1^2$.

Condition number of a matrix

Suppose $f(x) = \|Ax - b\|_2^2 = x^T A^T Ax - 2x^T Ab + \|b\|_2^2$. It is a convex function with gradient $\nabla f(x) = 2A^T Ax - 2Ab$.

Let $\sigma_{\max}(A) = \sup_{\|x\|_2=1} \|Ax\|_2$ and let $\sigma_{\min}(A) = \inf_{\|x\|_2=1} \|Ax\|_2$.

Definition

The condition number of A, denoted by $\kappa(A)$, is $\frac{\sigma_{\max}(A)}{\sigma_{\min}(A)}$.

Recall that $\lambda_{\min}(H(x))$ is the strong convexity parameter of f. For regression $2A^T A$ is the Hessian and hence $\lambda_{\min}(A^T A) = \sigma_{\min}^2$. Thus $\kappa(A) = L/\mu$ where L is the smoothness parameter and μ is the strong convexity parameter.

Gradient descent convergence when condition number is small

It is known that gradient descent converges very fast when L/μ is bounded. We state a lemma that captures this in a special case of regression while also making an additional assumption.

Lemma

Suppose all singular vectors of A are in the range $[1 - 1/\sqrt{2}, 1 + 1/\sqrt{2}]$. If we do gradient descent for regression with $\eta = 1/2$ then for all $t \geq 0$ we have

$$\|Ax^{(t+1)} - Ax^*\|_2 \leq 2^{-t}\|Ax^{(0)} - Ax^*\|_2$$

In other words the error of the vector $x^{(t)}$ after t steps goes down exponentially with t when compared to the initial error.

Gradient descent convergence when condition number is small

The lemma in the previous slide is a special case of a more general theorem about convergence of gradient descent for strongly convex functions. For a direct proof of the stated lemma for regression in previous slide see Nelson's notes.

Lemma

Suppose f is an L-smooth and μ-strongly convex function. Gradient descent with $\eta \leq 1/L$ satisfies the property that

$$\|x^{(t)} - x^*\|_2^2 \leq (1 - \alpha\mu)^t \|x^{(0)} - x^*\|_2^2.$$

Implication for Regression

Lemma shows that if condition number of A is small then gradient descent converges very fast. In pariticular if we have a good starting point $x^{(0)}$ such that $\|Ax^{(0)} - b\|_2 \leq c\|Ax^* - b\|$ for some constant c then gradient descent has the following property.

Lemma

After $t = O(\log(c/\epsilon))$ steps we have $\|Ax^{(t)} - b\|_2 \leq (1+\epsilon)\|Ax^* - b\|$.

To see this we observe via triangle inequality and lemma,

$\|Ax^{(t)} - b\|_2 \leq \|Ax^{(t)} - Ax^*\|_2 + \|Ax^* - b\|_2 \leq 2^{-t}\|Ax^{(0)} - Ax^*\|_2 + \|Ax^* - b\|_2.$

By triangle inequality

$\|Ax^{(0)} - Ax^*\|_2 \leq \|Ax^{(0)} - b\|_2 + \|Ax^* - b\|_2.$

Putting together

$\|Ax^{(t)} - b\|_2 \leq 2^{-t}\|Ax^{(0)} - b\|_2 + (1 + 2^{-t})\|Ax^* - b\|_2 \leq (1 + O(\epsilon))\|Ax^* - b\|_2$

Oblivious Subspace Embeddings Again

Suppose we use α-approximate oblivious subspace embedding for the columns $a_1, a_2, \ldots, a_d$ via a $k \times n$ sketch matrix Π. Thus we obtain $A' = \Pi A$. Previously we used $\alpha = (1+\epsilon)$ and solved the regression problem $\min_{x' \in \mathbb{R}^d} \|A'x' - b'\|_2$ where $b' = \Pi b$. This required k to be $\Theta(d/\epsilon^2)$. Now we instead use $\alpha = (1+\epsilon_0)$ for some fixed constaint ϵ_0 (say $1/4$).

Let $A' = \Pi A = U'\Sigma'(V')^T$ where we compute SVD of A'. Note U' is an orthonormal basis for the columns of A'. Let $R = V'(\Sigma')^{-1}$.

Claim

The singular values of AR are in the range $[1-\epsilon_0, 1+\epsilon_0]$.

Oblivious Subspace Embeddings Again

Claim

The singular values of AR are in the range $[1 - \epsilon_0, 1 + \epsilon_0]$.

To see this consider any vector z:

$$\|z\|_2 = \|U'z\|_2 = \|\Pi ARz\|_2 = (1 \pm \epsilon_0)\|ARz\|_2.$$

The first equality is from ortonormality of U', and second ineq is since Π is a $(1 + \epsilon_0)$-approximate OBSE.

Claim

The column space of A and AR are the same since V' is orthonormal and Σ' is a diagonal matrix.

Thus solving $\min_x \|Ax - b\|_2$ is same as solving $\min_y \|ARy - b\|_2$. If y^* is solution to latter problem then $x^* = Ry^*$ is a solution to the original problem.

Oblivious Subspace Embeddings Again

The previous two claims imply that gradient descent on $\boldsymbol{AR}$ will converge very fast since its condition number if small and moreover a solution to $\min_{\boldsymbol{y}} \|\boldsymbol{ARy} - \boldsymbol{b}\|_2$ allows us to recover a solution to the original regression problem with the same approximation quality.

Since Π is constant factor approximate OBSE, we can use the SVD $\boldsymbol{U}'\Sigma'(\boldsymbol{V}')^{\boldsymbol{T}}$ of $\Pi\boldsymbol{A}$ to obtain a constant factor approximate starting solution $\boldsymbol{x}^{(0)}$ to start the gradient descent. This implies that the number of iterations required for an eventual $(1+\epsilon)$-approximation is $O(\log(1/\epsilon))$. Each iteration requires computing $\boldsymbol{ARx}^{(\boldsymbol{t})}$.

Computing $\boldsymbol{ARx}^{(\boldsymbol{t})}$ can be done in $O(d^2 + \text{nnz}(\boldsymbol{A}))$ where $\text{nnz}(\boldsymbol{A})$ is the number of non-zeroes in $\boldsymbol{A}$.

Summarizing the algorithm

Input A, b where A is $n \times d$ matrix and $b \in \mathbb{R}^n$ with $n \geq d$

- Use $(1 + \epsilon_0)$-approximate OBSE embedding $k \times n$ matrix Π with $k = O(d)$ and compute $A' = \Pi A$ (use fast JL)
- Compute SVD $U'\Sigma'(V')^T$ of A' and let $R = V'(\Sigma')^{-1}$
- Use SVD to compute a good starting solution for $y^{(0)}$ for the problem $\min_y \|ARy - b\|_2$
- Use gradient descent for solving $\min_y \|ARy - b\|_2$ with starting solution $y^{(0)}$ and terminate in $t = O(\log(1/\epsilon))$ iterations
- Output $Ry^{(t)}$

We have reduced dependence on ϵ by using ϵ_0 approximate OBSE for some fixed ϵ_0 and then using gradient descent which has much better dependence on ϵ. For high accurate solutions this is an advantage.

Part I

Proof of GD Convergence for Strongly Convex Functions

Convergence of GD

Recall strong convexity implies that

$$f(y) \geq f(x) + (\nabla f(x))^T (y - x) + \frac{\mu}{2} \|y - x\|_2^2$$

We need a very useful lemma.

Lemma

Suppose f is μ-strongly convex then it also satisfies the Polyak-Lojasiewicz condition that $\|\nabla f(x)\|_2^2 \geq 2\mu(f(x) - f(x^*))$.

Intuition: strongly convex means function is has a strong curvature. Thus, the farther x is from x^* (where gradient is 0) the larger the gradient.

Properties from smoothness

Lemma

Suppose f is L-smooth. Then

1. $f(y) - f(x) - (\nabla f(x))^T(y - x) \leq \frac{L}{2}\|x - y\|_2^2$
2. $f(x - \frac{1}{L}\nabla f(x)) - f(x) \leq -\frac{1}{2L}\|\nabla f(x)\|_2^2$

Corollary

Suppose f is L-smooth then $\|\nabla f(x)\|_2^2 \leq 2L(f(x) - f(x^*))$.

Follows from part (2) of Lemma since

$$f(x^*) - f(x) \leq f(x - \frac{1}{L}\nabla f(x)) - f(x) \leq -\frac{1}{2L}\|\nabla f(x)\|_2^2$$

Properties from smoothness

Suppose f is L-smooth. Then
$f(y) - f(x) - (\nabla f(x))^T(y - x) \leq \frac{L}{2}\|x - y\|_2^2$.

Consider univariate function $g(\cdot)$ where
$g(t) = f(x + t(y - x)) - (\nabla f(x))^T(x + t(y - x))$. Note that
$g(0) = f(x) - (\nabla f(x))^T x$ and $g(1) = f(y) - (\nabla f(x))^T y$.

$$
\begin{aligned}
g(1) - g(0) &= \int_0^1 g'(t)dt = \int_0^1 (\nabla f(x + t(y - x)) - \nabla f(x))^T(y - x)d \\
&\leq \int_0^1 \|(\nabla f(x + t(y - x)) - \nabla f(x))\| \|(y - x)\| dt \\
&\leq \int_0^1 Lt\|y - x\|^2 dt = \frac{L}{2}\|y - x\|^2.
\end{aligned}
$$

We used smoothness to go from second to third line.

Properties from smoothness

Second part: $f(x - \frac{1}{L}\nabla f(x)) - f(x) \leq -\frac{1}{2L}\|\nabla f(x)\|_2^2$

Using first part with $y = x - \frac{1}{L}\nabla f(x)$,

$$f(x - \frac{1}{L}\nabla f(x)) - f(x) - (\nabla f(x))^T(-\frac{1}{L}\nabla f(x)) \leq \frac{L}{2}\|\frac{1}{L}\nabla f(x)\|_2^2$$

Simplifying and rearranging terms gives the desired property.

Polyak-Lojasiewicz condition

We don't need this but it is a nice contrast to the previous lemma.

Lemma

Suppose f is μ-strongly convex then it also satisfies the Polyak-Lojasiewicz condition that $\|\nabla f(x)\|_2^2 \geq 2\mu(f(x) - f(x^*))$.

Applying strong convexity with $y = x^*$ and rearranging

$$\begin{aligned} f(x) - f(x^*) &\leq (\nabla f(x))^T(x - x^*) - \frac{\mu}{2}\|x - x^*\|_2^2 \\ &= \frac{1}{2\mu}\|\nabla f(x)\|_2^2 - \frac{1}{2}\|\sqrt{\mu}(x - x^*) - \frac{1}{\sqrt{\mu}}\nabla f(x)\|_2^2 \\ &\leq \frac{1}{2\mu}\|\nabla f(x)\|_2^2. \end{aligned}$$

Rearranging gives the desired claim.

Proof of convergence of GD for strongly convex functions

Lemma

Suppose f is an L-smooth and μ-strongly convex function. Gradient descent with $\eta \leq 1/L$ satisfies the property that $\|x^{(t)} - x^*\|_2^2 \leq (1 - \alpha\mu)^t \|x^{(0)} - x^*\|_2^2$.

Suffices to prove the following

$$\|x^{(t+1)} - x^*\|_2^2 \leq (1 - \alpha\mu)\|x^{(t)} - x^*\|_2^2$$

and apply it repeatedly.

Proof contd

$$
\begin{aligned}
\|\boldsymbol{x}^{(t+1)} - \boldsymbol{x}^*\|_2^2 &= \|\boldsymbol{x}^{(t)} - \eta \nabla \boldsymbol{f}(\boldsymbol{x}^{(t)}) - \boldsymbol{x}^*\|_2^2 \quad \text{from GD algorithm} \\
&= \|\boldsymbol{x}^{(t)} - \boldsymbol{x}^*\|_2^2 - 2\eta(\nabla \boldsymbol{f}(\boldsymbol{x}^{(t)}))^{\boldsymbol{T}}(\boldsymbol{x}^{(t)} - \boldsymbol{x}^*) + \eta^2 \|\nabla \boldsymbol{f}(\boldsymbol{x}^{(t)})\|_2^2 \\
&\leq (1 - \alpha\mu)\|\boldsymbol{x}^{(t)} - \boldsymbol{x}^*\|_2^2 - 2\eta(\boldsymbol{f}(\boldsymbol{x}^{(t)}) - \boldsymbol{f}(\boldsymbol{x}^*)) + 2\eta^2 \boldsymbol{L}\|\nabla \boldsymbol{f}(\boldsymbol{x}^{(t)})\|_2^2 \quad \text{(strong convexity ineq)} \\
&\leq (1 - \alpha\mu)\|\boldsymbol{x}^{(t)} - \boldsymbol{x}^*\|_2^2 - 2\eta(\boldsymbol{f}(\boldsymbol{x}^{(t)}) - \boldsymbol{f}(\boldsymbol{x}^*)) + 2\eta^2 \boldsymbol{L}(\boldsymbol{f}(\boldsymbol{x}^{(t)}) - \boldsymbol{f}(\boldsymbol{x}^*)) \quad \text{(smoothness corollary} \\
&\leq (1 - \alpha\mu)\|\boldsymbol{x}^{(t)} - \boldsymbol{x}^*\|_2^2 - 2\eta(1 - \eta \boldsymbol{L})(\boldsymbol{f}(\boldsymbol{x}^{(t)}) - \boldsymbol{f}(\boldsymbol{x}^*)) \\
&\leq (1 - \alpha\mu)\|\boldsymbol{x}^{(t)} - \boldsymbol{x}^*\|_2^2 \quad \text{(since } \eta \leq 1/\boldsymbol{L} \text{ and } \boldsymbol{f}(\boldsymbol{x}^{(t)}) - \boldsymbol{f}(\boldsymbol{x}^*)\text{)}
\end{aligned}
$$