CS 15-859: Algorithms for Big Data Fall 2022

Lecture 1 — 2022/09/08

Prof. David Woodruff *Scribe: Yiwen (Victor) Song*

1 Regression Analysis (Part 1)

Definition. *Regression* is a statistical method to study dependencies between variables in the presence of noise. *Linear Regression* is a statistical method to study *linear* dependencies between variables in the presence of noise.

Example 1. Ohm's law gives $V = RI$. We can measure voltages V against currents I, and try to find a linear function that best fits the data. The slope would be R.

1.1 Standard Setting for Linear Regression

Definition. The assumption for linear regression is $b = x_0 + a_1x_1 + ... + a_dx_d + \epsilon$, where

- b is a measured variable
- a_i is a set of predictor variables
- x_i are model parameters we want to learn
- ϵ is noise

Note that we can assume $x_0 = 0$ because we can transform the above setting into one with $d + 1$ predictor variables where the first predictor variable is 1.

If we extend this setting to n observations of b and a_i, we can represent this setting in matrix form: $b = Ax$, where

- b is a $n \times 1$ vector of measured variables
- A is a $n \times d$ matrix of predictor variables where each row correspond to one observation
- x is a $d \times 1$ vector of model parameters we want to learn

We will be focusing on the over-constrained case when $n >> d$, i.e. there are many more examples than features.
Since we dropped the ϵ noise term, the goal is to find x^* such that Ax^* is closest to b. Using the least squares method, this can be concretely stated as:

$$min_x|Ax - b|_2^2 = min_x \sum(b_i - < A_i, x >)^2$$

Least squares method has some desirable statistical properties. If we assume the noise ϵ is Gaussian and independent across the examples, then the minimizer for the least squares regression gives the maximal likelihood estimator.

Looking at the geometry of this regression, if we rewrite $Ax = A_1x_1 + A_2x_2 + ... + A_dx_d$ where A_i is the i-th column of A, then we can see that Ax is in $Col(A)$ which is a d-dimensional subspace of R^n. Then we can reinterpret the problem as finding the projection of b onto $Col(A)$, i.e. computing the point on the column space of A that is nearest to b in l2-norm.

1.2 Solving Least Squares Regression via the Normal Equations

To solve the least squares regression, we can start by rewriting $b = Ax' + b'$, where b' is orthogonal to columns of A. This means that $< b', Ax_t >= 0$ for any x_t. Then the problem becomes

$$min_x|Ax - b|_2^2 = min_x|A(x - x') - b'|_2^2 = min_x|A(x - x')|_2^2 + |b'|_2^2 + 0 = min_x|A(x - x')|_2^2$$

So the optimal solution is $A(x - x') = 0$, which is equivalent to $A^TA(x - x') = 0$. This is true because for any x_t:

$$A^TAx_t = 0 \implies x_t^TA^TAx_t = 0 \implies (Ax_t)^TAx_t = 0 \implies |Ax_t|_2^2 = 0 \implies Ax_t = 0$$

Furthermore, $A^T(Ax - b) = A^T(A(x - x') - b') = A^TA(x - x')$ since $A^Tb = 0$.
Therefore, the optimal solution is

$$A^T(Ax - b) = 0 \Leftrightarrow A^TAx = A^Tb \tag{1}$$

If we assume that columns of A are linearly independent, then we have a closed-form solution $x = (A^TA)^{-1}A^Tb$. If columns of A are not linearly independent, the Moore-Penrose pseudoinverse gives a minimum norm solution x.

1.2.1 Moore-Penrose Pseudoinverse

Definition. Singular value decomposition (SVD) for a $n \times d$ matrix A gives

$$A = U\Sigma V^T$$

- U is $n \times d$ and has orthonormal columns
- Σ is a $d \times d$ diagonal matrix where $\Sigma_{11} \geq \Sigma_{22} \geq ... \geq \Sigma_{dd}$
- V is $d \times d$ and has orthonormal columns (i.e. V^T has orthonormal rows)

Definition. Pseudoinverse of A is

$$A^- = V\Sigma^{-1}U^T$$

Σ^{-1} is the diagonal matrix where

$$\Sigma_{ii}^{-1} = \begin{cases} \frac{1}{\Sigma_{ii}} & \Sigma_{ii} > 0 \\ 0 & \Sigma_{ii} = 0 \end{cases}$$

1.2.2 Using Pseudoinverse in the Normal Equations

Claim 1. $x = A^- b$ is an optimal solution to least squares regression

Proof. We need to show that $A^T Ax = A^T b$. Substituting with $x = A^- b$, we have the following

$$A^T Ax = A^T AA^- b = (V\Sigma U^T)(U\Sigma V^T)(V\Sigma^{-1}U^T)b = V\Sigma^2\Sigma^{-1}U^T b = V\Sigma U^T b = A^T b$$

This follows from the fact that $U^T U = V^T V = I$ and $\Sigma(\Sigma\Sigma^{-1}) = \Sigma$. The latter is true because $\Sigma\Sigma^{-1}$ is a diagonal matrix where

$$(\Sigma\Sigma^{-1})_{ii} = \begin{cases} 1 & \Sigma_{ii} > 0 \\ 0 & \Sigma_{ii} = 0 \end{cases}$$

which means that

$$(\Sigma(\Sigma\Sigma^{-1}))_{ii} = \begin{cases} \Sigma ii * 1 = \Sigma ii & \Sigma_{ii} > 0 \\ \Sigma ii * 0 = 0 & \Sigma_{ii} = 0 \end{cases}$$

■

Claim 2. $min_x |Ax - b|_2^2$ is not unique when columns of A are linearly dependent. In particular, any optimal solution has the form $x = A^- b + (I - V'V'^T)z$, where V'^T corresponds to the rows i of V^T for which $\Sigma_{ii} > 0$.

Proof. To better understand V'^T, suppose Σ has rank r, then V'^T is $r \times d$. Let V''^T corresponds to rest of the rows in V^T, i.e. the rows i of V^T for which $\Sigma_{ii} = 0$. Then we have

$$A = U\Sigma V^T = U \begin{bmatrix} \Sigma_{11} & & & & & \\ & \ddots & & & & \\ & & \Sigma_{rr} & & & \\ & & & 0 & & \\ & & & & \ddots & \\ & & & & & 0 \end{bmatrix} \begin{bmatrix} & \\ & \\ V'^T & \\ & \\ & \\ V''^T & \\ & \end{bmatrix}$$

We need to show $A^T Ax = A^T b \Leftrightarrow A^T A(A^- b + (I - V'V'^T)z) = A^T b \Leftrightarrow A^T A(I - V'V'^T)z) = 0$.
It suffices to show that $A(I - V'V'^T)z = 0$ for any z.
This is true because $(I - V'V'^T)$ is a projection onto the kernel space of A.
Therefore, $x = A^- b + (I - V'V'^T)z$ is a solution for any z. Since this is a $(d - r)$-dimensional affine space, it spans all optimal solutions. ■

Claim 3. Among all solutions $x = A^- b + (I - V'V'^T)z$, $x = A^- b$ has minimum norm.

Proof.

$$A^-b = V\Sigma^{-1}U^Tb = \begin{bmatrix} V' & V'' \end{bmatrix} \begin{bmatrix} \frac{1}{\Sigma_{11}} & & & & & \\ & \ddots & & & & \\ & & \frac{1}{\Sigma_{rr}} & & & \\ & & & 0 & & \\ & & & & \ddots & \\ & & & & & 0 \end{bmatrix} U^Tb$$

Notice that we are scaling columns of V' by $\frac{1}{\Sigma_{ii}}$, and columns of V' are scaled by 0.
So A^-b is in $Col(V')$.
Let $A^-b = V'y$ for some y. Then

$$(A^-b)^T(I - V'V'^T)z = y^TV'^T(I - V'V'^T)z = y^T(V'^T - V'TV'V'^T)z = y^T(V'^T - V'^T)z = 0$$

Therefore, $A^-b \perp (I - V'V'^T)z$.
Then by the Pythagorean theorem, $|A^-b + (I - V'V'^T)z|_2^2 = |A^-b|_2^2 + |(I - V'V'^T)z|_2^2 \geq |A^-b|_2^2$. ■

1.2.3 Time Complexity

Since we need to compute $x = A^-b$, naively this takes $O(nd^2)$ time. Using fast matrix multiplication, we can reduce this to $O(nd^{1.376})$. But we still want much better running time.

1.3 Sketching to Solve Least Squares Regression

We have a much more efficient algorithm that finds an approximate solution x to $min_x|Ax - b|_2$. The following sketching algorithm outputs x' for which $|Ax' - b|_2 \leq (1 + \epsilon)min_x|Ax - b|_2$ with high probability:

1. Draw S from a $k \times n$ random family of matrices ($k << n$)
2. Compute SA and Sb
3. Output the solution x' to $min_{x'}|(SA)x - Sb|_2$, which is $x' = (SA)^-Sb$.

This algorithm takes $O(kd^2)$ time, or $O(kd^{1.376})$ with fast matrix multiplication. Since $k << n$, this is a major improvement.

In particular, we choose S to be a $\frac{d}{\epsilon^2} \times n$ matrix of i.i.d. Normal random variables sampled from $N(0, \frac{1}{k})$. In the next notes, we will introduce subspace embedding to show why this works.