회귀분석(Regression Analysis)

이정우
phyjics@gmail.com

2019년 9월 16일

차 례

제 1 장

최소 제곱법(Least Squares Fitting)

1.1 선형 회귀 - 직선

N개의 데이터 점(x_i, y_i)이 직선

$$f(x) = ax + b, \tag{1.1}$$

의 분포를 따른다고 할 때 최소 제곱법은 데이터와 직선 사이의 y값 차이를 제곱하여 합한 값을 최소화 하는 방법이다. 최소화 하고자 하는 값 S는 다음과 같다.

$$S = \sum_{i=1}^{N} (y_i - ax_i - b)^2. \tag{1.2}$$

이를 변수 a와 b에 대해서 각각 최소화 하면;

$$\frac{\partial S}{\partial a} = 0, \quad \frac{\partial S}{\partial b} = 0, \tag{1.3}$$

아래와 같이 a와 b를 구할 수 있다.

$$a = \frac{\langle x_i y_i \rangle - \langle x_i \rangle \langle y_i \rangle}{\langle x_i^2 \rangle - \langle x_i \rangle^2}. \tag{1.4}$$

$$b = \frac{\langle x_i^2 \rangle \langle y_i \rangle - \langle x_i \rangle \langle x_i y_i \rangle}{\langle x_i^2 \rangle - \langle x_i \rangle^2}. \tag{1.5}$$

1.2 함수를 알고 있는 경우

N개의 데이터 점(x_i, y_i)이

$$f(x) = av(x - b), \tag{1.6}$$

의 분포를 따른다고 할 때 최소화 하고자 하는 S는

$$S = \sum_{i=1}^{N} \left(y_i - av(x_i - b) \right)^2, \tag{1.7}$$

이 된다. 이를 a와 b에 대해서 최소화 하면 $(\partial S/\partial a = 0,\ \partial S/\partial b = 0)$ 다음과 같다.

$$\frac{\partial S}{\partial a} = 0 \quad \longrightarrow \quad a = \frac{\sum y_i v(x_i - b)}{\sum f^2(x_i - b)}. \tag{1.8}$$

함수의 형태를 알지 못할 때 변수 b는 함수 v에 종속되어 있어 구할 수 없다. 하지만 특정 b에서 최소 제곱법에 맞는 a를 구할 수 있다. 함수의 형태를 알고 있을 때는 경우에 따라 다음 식을 이용하여 b를 구할 수 있다.

$$\frac{\partial s}{\partial b} = 0 \quad \longrightarrow \quad a = \frac{\sum y_i v'(x_i - b)}{\sum v(x_i - b) v'(x_i - b)} \tag{1.9}$$

$$v'(x) \equiv \frac{\partial v(x)}{\partial x} \tag{1.10}$$

제 2 장

원 피팅

2.1 리만 피팅

2.1.1 극사영

극사영은(stereographic projection)은 구면의 점을 평면으로, 또는 반대로 옮기는 함수이며 입체사영(conformal mapping)으로도 불린다. (x, z) 평면위의 점 $\mathrm{P}_i(x_i, 0, z_i)$와, 원점 $\mathrm{O}(0, 0, 0)$ 위에 놓인 리만 구(Riemann sphere) S를 생각해보자. 이 구는 R의 반지름을 가지며 중심점 C는 $(0, R, 0)$로 주어진다.

$$\mathrm{S}\ :\ x^2 + (y - R)^2 + z^2 = R^2 \tag{2.1}$$

점 P_i에서 리만 구의 꼭지점 $\mathrm{T}(0, 2R, 0)$를 잇는 선이 구와 만나는 점을 $\mathrm{P}'_i(x'_i, y'_i, z'_i)$라고 하자. 반지름 r_i를 $r_i \equiv \sqrt{x_i^2 + z_i^2}/2R$ 와 같이 정의 하였을 때 점 P'_i는 점 P_i로 부터 다음과 같이 주어진다.

$$\begin{aligned} x'_i &= \frac{x_i}{1 + r_i^2} \\ y'_i &= \frac{2Rr_i^2}{1 + r_i^2} \\ z'_i &= \frac{z_i}{1 + r_i^2} \end{aligned} \tag{2.2}$$

반대로 점 P_i는 점 P'_i로 부터 다음과 같이 주어진다.

$$\begin{aligned} x_i &= \frac{x'_i}{1 - y'_i/2R} \\ z_i &= \frac{z'_i}{1 - y'_i/2R} \end{aligned} \tag{2.3}$$

2.1.2 리만 구 위의 원

리만 구 위에 사영된 데이터 점 P'_i을 평면으로 피팅을 하였을 때(3.1절 참조) 구해진 평면 K의 수직 벡터를 $\mathbf{n}(a, b, c)$, 평면에 속한 한점(P'_i의 평균)을 $\mathrm{M}(\langle x\rangle', \langle y\rangle', \langle z\rangle')$이라 하자.

$$\mathrm{K}\ :\ a(x - \langle x\rangle') + b(y - \langle y\rangle') + c(z - \langle z\rangle') = 0 \tag{2.4}$$

평면이 구를 자르면서 생기는 원의 중심점 A는 리만구의 중심점 C를 지나고 **n**의 방향을 가진 직선 L이 평면 K와 만나는 점이 된다.

$$\mathrm{L}\ :\ \begin{cases} x = at \\ y = bt + R \\ z = ct \end{cases} \tag{2.5}$$

연립하여 풀면 중심점에서의 t인 t_A는 다음과 같고

$$t_A = \frac{a\langle x\rangle' + b(\langle y\rangle' - R) + c\langle z\rangle'}{a^2 + b^2 + c^2}, \tag{2.6}$$

원의 중심점 $\mathrm{C_A}$는 $(at_A, bt_A + R, ct_A)$이 되며 리만 구의 중심에어 원의 중심까지의 거리가 $l = |\overline{\mathrm{CC_A}}|$ 일때 원의 반지름 r_A는 $\sqrt{R^2 - l^2}$이 된다.

2.1.3 역 극사영

원 A를 xz평면 위에 눕고$(\mathrm{A}_\perp, \mathbf{x}_\perp)$

$$\mathrm{A}_\perp\ :\ x_\perp^2 + z_\perp^2 = r_A^2 \tag{2.7}$$

벡터 $\hat{\mathbf{y}}$가 피팅한 평면의 수직벡터$\mathbf{n}$과 같도록 회전시킨 후 y 방향으로 $(\mathrm{C_A})_y$ 만큼 올린 원이 된다. 회전 행렬 $\mathbf{R}$을 x축, y축으로 분리해서 보자.

$$\mathbf{R} = \mathbf{R_y}(\phi)\mathbf{R_x}(\theta), \tag{2.8}$$

$$\mathbf{R_y}(\phi) = \begin{pmatrix} \cos\phi & 0 & \sin\phi \\ 0 & 1 & 0 \\ -\sin\phi & 0 & \cos\phi \end{pmatrix},\ \mathbf{R_x}(\theta) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{pmatrix} \tag{2.9}$$

이 회전은 $\hat{\mathbf{y}}$가 $\mathbf{n}$가 되는 회전과 같다.

$$\mathbf{x}' = \mathbf{R}\mathbf{x}_\perp \quad \longrightarrow \quad \mathbf{R}\hat{\mathbf{y}} = \mathbf{n}, \tag{2.10}$$

따라서 다음과 같이 행렬의 각 성분들을 구할 수 있다.

$$\begin{aligned} \cos\theta &= b \\ \sin\theta &= d \\ \cos\phi &= c/d \\ \sin\phi &= a/d \end{aligned} \quad , \quad d \equiv \sqrt{1 - b^2} \tag{2.11}$$

이 되고 $\mathbf{R}$, $\mathbf{R}^{-1}$과 변환 식은 다음과 같이 쓸 수 있다.

$$\mathbf{R} = \begin{pmatrix} c/d & a & ab/d \\ 0 & b & -d \\ -a/d & c & bc/d \end{pmatrix},\ \mathbf{R}^{-1} = \begin{pmatrix} c/d & 0 & -a/d \\ a & b & c \\ ab/d & -d & bc/d \end{pmatrix} \tag{2.12}$$

$$\mathbf{x}_\perp \to \mathbf{x}' \quad \begin{cases} x' = (c/d)x_\perp + ay_\perp + (ab/d)z_\perp \\ y' = by_\perp - dz_\perp \\ z' = -(a/d)x_\perp + cy_\perp + (bc/d)z_\perp \end{cases} \tag{2.13}$$

$$\mathbf{x}' \to \mathbf{x}_\perp \quad \begin{cases} x_\perp = (c/d)x' - (a/d)z' \\ y_\perp = ax' + by' + cz' \\ z_\perp = (ab/d)x' - dy' + (bc/d)z' \end{cases} \tag{2.14}$$

원 $\mathrm{A}_\perp$를 변환식을 통하여 치환하고 y 방향으로 $(\mathrm{C_A})_y$ 만큼 올리면 다음과 같다.

$$(cx' - az')^2 + \left(abx' - d^2(y' - (\mathrm{C_A})_y) + bcz'\right)^2 = r_A^2 d^2 \tag{2.15}$$

다음으로 식 (2.2) 을 이용하여 리만구의 원을 xz 평면으로 옮겨 보자.

$$(cx - az)^2 + \left(abx - d^2\left(2Rr^2 - (1+r^2)(\mathrm{C_A})_y\right) + bcz\right)^2 = \left(r_A d(1+r^2)\right)^2 \tag{2.16}$$

그런데 이렇게 계산하다보면 끝이 나지 않는다! 이럴수가! 살려줘!

2.1.4 역 극사영 초콜릿 먹고싶다

작전은 이렇다. 원 A에서 y 값이 가장 작은 점과 가장 큰 점을 구해서 역 극사영 시킨다. 이 두점은 역 극사영 된 후에도 서로 원의 정 반대편에 위치하므로 xz 평면에서의 최종 원을 쉽게 구할 수 있다. 중심점은 두 점의 중간지점이고 반지름은 두 점 사이 거리의 반이다! 오! 된다! 코딩해봤더니 됩니다!

$$\mathbf{R}' = \mathbf{R_y}(\phi)\mathbf{R_x}(\alpha), \tag{2.17}$$

$$\mathbf{R_y}(\phi) = \begin{pmatrix} \cos\phi & 0 & \sin\phi \\ 0 & 1 & 0 \\ -\sin\phi & 0 & \cos\phi \end{pmatrix}, \quad \mathbf{R_x}(\alpha) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{pmatrix} \tag{2.18}$$

$$\begin{aligned} \cos\alpha &= l/R \\ \sin\alpha &= r_A/R \end{aligned} \tag{2.19}$$

2.2 간단한 원 피팅

가중치 w_i를 가지고 있는 N개의 데이터 점(x_i, y_i)이 원

$$(x - x_c)^2 + (y - y_c)^2 = r^2, \tag{2.20}$$

의 분포를 따른다고 하자. 원의 반지름 r 제곱과 원의 중심에서 데이터 점 까지의 거리 제곱의 차이의 제곱을 모두 합한 값를 최소화 해보자. 즉, 다음 S를 최소화 한다.

$$\Lambda_i(x_c, y_c, r) = r^2 - \left[(x_i - x_c)^2 + (y_i - y_c)^2\right] \tag{2.21}$$

$$S(x_c, y_c, r) = \sum_{i=1}^{N} w_i \Lambda_i^2 \tag{2.22}$$

이때 식 (2.21)의 x와 y는 데이터의 기대값 만큼 평행 이동한 $u(=x-\langle x\rangle)$와 $v(=y-\langle y\rangle)$로 다시 쓸 수 있다.

$$
\begin{aligned}
\Lambda_i(u_c, v_c, r) &= r^2 - \left[(x_i - x_c)^2 + (y_i - y_c)^2\right] &(2.23)\\
&= r^2 - \left[(x_i - \langle x\rangle - x_c + \langle x\rangle)^2 + (y_i - \langle y\rangle - y_c + \langle y\rangle)^2\right]\\
&= r^2 - \left[((x_i - \langle x\rangle) - (x_c - \langle x\rangle))^2 + ((y_i - \langle y\rangle) - (y_c - \langle y\rangle))^2\right]\\
&= r^2 - \left[(u_i - u_c)^2 + (v_i - v_c)^2\right] &(2.24)\\
&= r^2 - u_i^2 - u_c^2 + 2u_iu_c - v_i^2 - v_c^2 + 2v_iv_c.
\end{aligned}
$$

이제 각 변수에 대해서 최소화 해보자.

$$\frac{\partial S}{\partial r} = 0 \quad \longrightarrow \quad \sum_i^N w_i\Lambda_i = 0 \tag{2.25}$$

$$\frac{\partial S}{\partial u_c} = 0 \quad \longrightarrow \quad \sum_i^N w_iu_i\Lambda_i = 0 \tag{2.26}$$

$$\frac{\partial S}{\partial v_c} = 0 \quad \longrightarrow \quad \sum_i^N w_iv_i\Lambda_i = 0 \tag{2.27}$$

식 (2.25)를 W로 나눈 후 풀어쓰면 다음과 같다.

$$
\begin{aligned}
\frac{1}{W}\sum_i^N w_i\Lambda_i &= \frac{1}{W}\sum_i^N w_i\left(r^2 - u_i^2 - u_c^2 + 2u_iu_c - v_i^2 - v_c^2 + 2v_iv_c\right) &(2.28)\\
&= r^2 - \langle u^2\rangle - u_c^2 + 2u_c\langle u\rangle - \langle v^2\rangle - v_c^2 + 2v_c\langle v\rangle\\
&= r^2 - (u_c^2 + v_c^2) - \langle u^2\rangle - \langle v^2\rangle = 0
\end{aligned}
$$

위 계산은 $\langle u\rangle = 0$, $\langle v\rangle = 0$ 임을 이용하였다. 위 식을 이용하면 반지름의 제곱은 다음과 같이 구해진다.

$$r^2 = u_c^2 + v_c^2 + \langle u^2\rangle + \langle v^2\rangle. \tag{2.29}$$

식 (2.26)과 식 (2.27)은 다음과 같이 나타난다.

$$
\begin{aligned}
\frac{1}{W}\sum_i^N w_iu_i\Lambda_i &= \frac{1}{W}\sum_i^N w_iu_i\left(r^2 - u_i^2 - u_c^2 + 2u_iu_c - v_i^2 - v_c^2 + 2v_iv_c\right) &(2.30)\\
&= r^2\langle u\rangle - \langle u^3\rangle - u_c^2\langle u\rangle + 2u_c\langle u^2\rangle - \langle uv^2\rangle - v_c^2\langle u\rangle + 2v_c\langle uv\rangle\\
&= -\langle u^3\rangle + 2u_c\langle u^2\rangle - \langle uv^2\rangle + 2v_c\langle uv\rangle = 0\\
\frac{1}{W}\sum_i^N w_iv_i\Lambda_i &= -\langle u^2v\rangle + 2u_c\langle uv\rangle - \langle v^3\rangle + 2v_c\langle v^2\rangle = 0 &(2.31)
\end{aligned}
$$

즉, 다음과 같은 연립 방정식이 나오며

$$
\begin{aligned}
2\langle u^2\rangle u_c + 2\langle uv\rangle v_c &= \langle u^3\rangle + \langle uv^2\rangle &(2.32)\\
2\langle uv\rangle u_c + 2\langle v^2\rangle v_c &= \langle u^2v\rangle + \langle v^3\rangle &(2.33)
\end{aligned}
$$

행렬 식으로 쓰면,

$$\begin{pmatrix} \langle u^2\rangle & \langle uv\rangle \\ \langle uv\rangle & \langle v^2\rangle \end{pmatrix}\begin{pmatrix} u_c \\ v_c \end{pmatrix} = \frac{1}{2}\begin{pmatrix} \langle u^3\rangle + \langle uv^2\rangle \\ \langle u^2v\rangle + \langle v^3\rangle \end{pmatrix} \tag{2.34}$$

$$\begin{pmatrix} u_c \\ v_c \end{pmatrix} = \frac{1}{2} \frac{1}{\langle u^2\rangle\langle v^2\rangle - \langle uv\rangle^2} \begin{pmatrix} \langle v^2\rangle & -\langle uv\rangle \\ -\langle uv\rangle & \langle u^2\rangle \end{pmatrix} \begin{pmatrix} \langle u^3\rangle + \langle uv^2\rangle \\ \langle u^2 v\rangle + \langle v^3\rangle \end{pmatrix} \tag{2.35}$$

와 같고 풀어쓰면 다음과 같다.

$$u_c = \frac{1}{2} \frac{\langle v^2\rangle \left(\langle u^3\rangle + \langle uv^2\rangle\right) - \langle uv\rangle \left(\langle u^2 v\rangle + \langle v^3\rangle\right)}{\langle u^2\rangle\langle v^2\rangle - \langle uv\rangle^2} \tag{2.36}$$

$$v_c = \frac{1}{2} \frac{\langle u^2\rangle \left(\langle u^2 v\rangle + \langle v^3\rangle\right) - \langle uv\rangle \left(\langle u^3\rangle + \langle uv^2\rangle\right)}{\langle u^2\rangle\langle v^2\rangle - \langle uv\rangle^2} \tag{2.37}$$

데이터가 완벽한 직선을 이룬다면 $\langle u^2\rangle\langle v^2\rangle - \langle uv\rangle^2 = 0$ 이 되면서 해가 없지만, 이 경우를 제외하면 원의 중심은 다음과 같이 주어진다.

$$x_c = \frac{1}{2} \frac{\langle v^2\rangle \left(\langle u^3\rangle + \langle uv^2\rangle\right) - \langle uv\rangle \left(\langle u^2 v\rangle + \langle v^3\rangle\right)}{\langle u^2\rangle\langle v^2\rangle - \langle uv\rangle^2} + \langle x\rangle \tag{2.38}$$

$$y_c = \frac{1}{2} \frac{\langle u^2\rangle \left(\langle u^2 v\rangle + \langle v^3\rangle\right) - \langle uv\rangle \left(\langle u^3\rangle + \langle uv^2\rangle\right)}{\langle u^2\rangle\langle v^2\rangle - \langle uv\rangle^2} + \langle y\rangle \tag{2.39}$$

제곱평균제곱근(Root Mean Square)은 그냥 계산...

$$\mathrm{RMS} = \sqrt{\frac{\sum_i^N w_i \Lambda_i^2}{\langle w\rangle_N (N-3)}} \tag{2.40}$$

제 3 장

수직 거리 회귀(Orthogonal Distance Regression (ODR))를 이용한 최소 제곱법

3.1 평면 피팅

3차원에서 주어진 N개의 점 데이터를 이용하여 가장 잘 맞는 평면을 피팅해보자. 이 문서에서 다룰 평면 피팅은 ODR(수직 거리 회귀, Orthogonal Distance Regression) 피팅으로 각 점에서 평면까지 수직 거리 제곱의 합을 최소화 하는 방법이다. 일반적인 피팅 방법에서는 문제에서 최소화 할 S를 정한 뒤 변수를 바꿔가면서 S가 최소가 되는 점을 찾아 나가는 반복 계산을 한다. 하지만 ODR 피팅은 S가 최소가 되는 지점을 3차원 행렬의 고윳값 문제로 해결 할 수 있다는 것이 증명되어 있다. 또한 이 방법으로 평면 피팅 뿐만 아니라 선 피팅도 할 수 있다.

N개의 점 데이터를 평면($ax+by+cz+d=0$)으로 피팅을 하고 싶다. i번째 점의 가중치는 w_i이며 위치는 (x_i, y_i, z_i)로 주어진다. 평면과 i번째 점 사이의 수직 거리는 다음과 같다.

$$D_i(a,b,c,d) = \frac{|ax_i + by_i + cz_i + d|}{\sqrt{a^2+b^2+c^2}}. \tag{3.1}$$

N개의 점에 대해서 가중치를 고려한 거리 제곱 D_i^2의 합은 다음과 같다.

$$\begin{aligned} S(a,b,c,d) &= \sum_{i=1}^{N} w_i\, D_i(a,b,c,d)^2 \qquad (3.2) \\ &= \sum_{i=1}^{N} w_i \frac{|ax_i + by_i + cz_i + d|^2}{a^2+b^2+c^2}. \end{aligned}$$

S를 변수 d에 대해서 최소화 하면 $\partial S/\partial d = 0$ 이므로 다음을 얻는다.

$$d = -\Big(a\langle x\rangle_N + b\langle y\rangle_N + c\langle z\rangle_N\Big). \tag{3.3}$$

여기서 $\langle x\rangle_N, \langle y\rangle_N, \langle z\rangle_N$는 N개 데이터에 대한 기댓값이고

$$\langle x\rangle_N \equiv \frac{1}{W_N}\sum_{i=1}^{N} w_i\, x_i \tag{3.4}$$

$$\langle y\rangle_N \equiv \frac{1}{W_N}\sum_{i=1}^{N} w_i\, y_i \tag{3.5}$$

$$\langle z\rangle_N \equiv \frac{1}{W_N}\sum_{i=1}^{N} w_i\, z_i \tag{3.6}$$

W는 가중치의 합이다.

$$W_N = \sum_{i=1}^{N} w_i. \tag{3.7}$$

식 (3.3)을 평면의 방정식에 대입하면 다음과 같다.

$$a\big(x_i - \langle x\rangle_N\big) + b\big(y_i - \langle y\rangle_N\big) + c\big(z_i - \langle z\rangle_N\big) = 0. \tag{3.8}$$

즉, 우리가 구하려고 하는 평면은 반드시 데이터의 중심점을 지나간다는 사실을 알 수 있다. 나아가 식 (3.2)는

$$\begin{aligned} S(a,b,c) &= \sum_{i=1}^{N} w_i \frac{|ax_i + by_i + cz_i - a\langle x\rangle_N - b\langle y\rangle_N - c\langle z\rangle_N|^2}{a^2+b^2+c^2} \\ &= \sum_{i=1}^{N} \frac{|a\sqrt{w_i}(x_i - \langle x\rangle_N) + b\sqrt{w_i}(y_i - \langle y\rangle_N) + c\sqrt{w_i}(z_i - \langle z\rangle_N)|^2}{a^2+b^2+c^2} \\ &= \sum_{i=1}^{N} \frac{|aX_{N,i} + bY_{N,i} + cZ_{N,i}|^2}{a^2+b^2+c^2} \end{aligned} \tag{3.9}$$

이 된다. $X_{N,i}$, $Y_{N,i}$, $Z_{N,i}$는 다음과 같이 정의한다.

$$X_{N,i} \equiv \sqrt{w_i}\,(x_i - \langle x\rangle_N) \tag{3.10}$$

$$Y_{N,i} \equiv \sqrt{w_i}\,(y_i - \langle y\rangle_N) \tag{3.11}$$

$$Z_{N,i} \equiv \sqrt{w_i}\,(z_i - \langle z\rangle_N) \tag{3.12}$$

이제 S를 행렬식으로 쓰면

$$\mathbf{v^T} \equiv (a, b, c,), \tag{3.13}$$

$$\mathbf{M} \equiv \begin{pmatrix} X_1 & Y_1 & Z_1 \\ X_2 & Y_2 & Z_2 \\ X_3 & Y_3 & Z_3 \\ \vdots & \vdots & \vdots \\ X_N & Y_N & Z_N \end{pmatrix}, \tag{3.14}$$

일때 다음과 같이 다시 쓸 수 있다.

$$S(\mathbf{v}) = \frac{\mathbf{v^T M^T M v}}{\mathbf{v^T v}} = \frac{\mathbf{v^T A v}}{\mathbf{v^T v}}. \tag{3.15}$$

$$\mathbf{A} \equiv \mathbf{M^T M} \tag{3.16}$$

$S(\mathbf{v})$는 Rayleigh Quotient로 행렬 $\mathbf{A}$의 가장 작은 고윳값(eigen value)을 택하였을때 최소화 되며, 반대로 가장 큰 고윳값을 택하였을때 최대화 된다. 이때 고유벡터(eigen vector) $\mathbf{v}$는 평면의 수직 벡터가 된다.

행렬 $\mathbf{M}$에 대해서 특이값 분해(Singular value decomposition)를 하고 이를 행렬$\mathbf{A}$에 대입하면

$$\mathbf{M} = \mathbf{USV^T} \tag{3.17}$$

$$\mathbf{A} = \mathbf{VS^2V^T} \tag{3.18}$$

가 된다. 즉 행렬 $\mathbf{A}$의 고윳값은 행렬 $\mathbf{M}$의 특이값의 제곱이 되며 행렬 $\mathbf{A}$의 고유벡터는 행렬 $\mathbf{M}$의 특이벡터와 같다. 여기서 특이값을 구하거나 고윳값을 구하므로써 평면을 구할 수 있는데 아직 특이값에 대해 자세히 알지 못하므로 어느 방법이 더 좋은지 알 수 없다(추가 바람). 따라서 지금은 행렬 $\mathbf{A}$의 고윳값 문제로 해결하도록 한다.

식 (3.18)에 따라 행렬 $\mathbf{A}$를 N의 함수로 풀어 쓰면 다음과 같다.

$$\mathbf{A}_N = \sum_{i=1}^{N} \begin{pmatrix} X_{N,i}^2 & X_{N,i}\,Y_{N,i} & X_{N,i}\,Z_{N,i} \\ Y_{N,i}\,X_{N,i} & Y_{N,i}^2 & Y_{N,i}\,Z_{N,i} \\ Z_{N,i}\,X_{N,i} & Z_{N,i}\,Y_{N,i} & Z_{N,i}^2 \end{pmatrix}. \tag{3.19}$$

이 행렬은 W_N로 나누었을때 공분산 행렬(covariance matrix)이 된다.

$\mathbf{A}_N$의 가장 작은 고윳값을 λ_{min}라고 할때 S의 최솟값은 바로 λ_{min}가 됨을 식 (3.15)으로 부터 알 수 있다. 또한 최소제곱의 제곱근인 RMS는 다음과 같다.

$$\mathrm{RMS} = \sqrt{\frac{\lambda_{min}}{\langle w\rangle_N (N-2)}} \tag{3.20}$$

λ_{min}를 W_N 가 아닌 $(W_N - 2\langle w\rangle_N)$로 나누는 이유는 평면을 만드는데 필요한 자유도가 2개 소모 되기 때문이다. 모든 가중치가 1인 경우에는 $(N-2)$로 나누어야 하지만 이 경우에는 자유도 하나 당 가중치의 평균을 빼주기 때문에 $(W_N - 2\,W_N/N)$로 나눈다. 즉, $\langle w\rangle_N (N-2)$으로 나눈다.

3.2 가만있어봐라

행렬 $\mathbf{A}$의 성분을 정리하면 다음과 같다.

$$\begin{aligned} (\mathbf{A}_N)_{11} &= \sum_{i=1}^{N} X_{N,i}^2 \\ &= \sum_{i=1}^{N} w_i\,(x_i - \langle x\rangle_N)^2 \\ &= \sum_{i=1}^{N} w_i\,(x_i^2 - 2x_i\langle x\rangle_N + \langle x\rangle_N^2) \\ &= \sum_{i=1}^{N} w_i\,x_i^2 - 2\langle x\rangle_N \sum_{i=1}^{N} w_i\,x_i + \langle x\rangle_N^2 \sum_{i=1}^{N} w_i \\ &= W_N\,\langle x^2\rangle_N - 2W_N\,\langle x\rangle_N^2 + W_N\,\langle x\rangle_N^2 \\ &= W_N\Big(\langle x^2\rangle_N - \langle x\rangle_N^2\Big). \end{aligned} \tag{3.21}$$

$$
\begin{aligned}
(\mathbf{A}_N)_{12} &= \sum_{i=1}^{N} X_{N,i} Y_{N,i} &(3.22)\\
&= \sum_{i=1}^{N} w_i\,(x_i - \langle x\rangle_N)(y_i - \langle y\rangle_N)\\
&= \sum_{i=1}^{N} w_i\,(x_i\,y_i + \langle y\rangle_N \langle x\rangle_N - x_i \langle y\rangle_N - y_i \langle x\rangle_N)\\
&= \sum_{i=1}^{N} w_i\,x_i\,y_i + \langle x\rangle_N\,\langle y\rangle_N \sum_{i=1}^{N} w_i - \langle y\rangle_N \sum_{i=1}^{N} w_i\,x_i - \langle x\rangle_N \sum_{i=1}^{N} w_i\,y_i\\
&= W_N \langle xy\rangle_N + W_N \langle x\rangle_N\,\langle y\rangle_N - W_N \langle y\rangle_N \langle x\rangle_N - W_N \langle x\rangle_N \langle y\rangle_N\\
&= W_N\Big(\langle xy\rangle_N - \langle x\rangle_N \langle y\rangle_N\Big).
\end{aligned}
$$

행렬 $\mathbf{A}$는 대칭 행렬이며 각 행에 따라서 x, y, z의 순서만 바뀌기 때문에 위 두 성분을 이용하여 다른 성분들 쉽게 알 수 있다.

이제 기존 데이터에서 다른 데이터를 추가할 때 행렬 $\mathbf{A}$가 어떻게 변하는지 알아보자. 각 성분들은 다음과 같다.

$$
\begin{aligned}
(\mathbf{A}_{N+1})_{11} &= W_{N+1}\Big(\langle x^2\rangle_{N+1} - \langle x\rangle_{N+1}^2\Big), &(3.23)\\
(\mathbf{A}_{N+1})_{22} &= W_{N+1}\Big(\langle y^2\rangle_{N+1} - \langle y\rangle_{N+1}^2\Big), &(3.24)\\
(\mathbf{A}_{N+1})_{33} &= W_{N+1}\Big(\langle z^2\rangle_{N+1} - \langle z\rangle_{N+1}^2\Big), &(3.25)\\
(\mathbf{A}_{N+1})_{12} &= W_{N+1}\Big(\langle xy\rangle_{N+1} - \langle x\rangle_{N+1} \langle y\rangle_{N+1}\Big), &(3.26)\\
(\mathbf{A}_{N+1})_{13} &= W_{N+1}\Big(\langle xz\rangle_{N+1} - \langle x\rangle_{N+1} \langle z\rangle_{N+1}\Big), &(3.27)\\
(\mathbf{A}_{N+1})_{23} &= W_{N+1}\Big(\langle yz\rangle_{N+1} - \langle y\rangle_{N+1} \langle z\rangle_{N+1}\Big). &(3.28)
\end{aligned}
$$

각 성분들을 이루는 기댓값들은 다음과 같이 기존 기댓값을 이용하여 구할 수 있다.

$$
\begin{aligned}
\langle x\rangle_{N+1} &= \frac{1}{W_{N+1}} \sum_{i=1}^{N+1} w_i\,x_i &(3.29)\\
&= \frac{1}{W_{N+1}} \left(\sum_{i=1}^{N} w_i\,x_i + w_{N+1}\,x_{N+1}\right)\\
&= \frac{1}{W_{N+1}} \Big(W_N \langle x\rangle_N + w_{N+1}\,x_{N+1}\Big).
\end{aligned}
$$

$$
\begin{aligned}
\langle x^2\rangle_{N+1} &= \frac{1}{W_{N+1}} \sum_{i=1}^{N+1} w_i\,x_i^2 &(3.30)\\
&= \frac{1}{W_{N+1}} \left(\sum_{i=1}^{N} w_i\,x_i^2 + w_{N+1}\,x_{N+1}^2\right)\\
&= \frac{1}{W_{N+1}} \Big(W_N \langle x^2\rangle_N + w_{N+1}\,x_{N+1}^2\Big).
\end{aligned}
$$

$$
\begin{aligned}
\langle xy\rangle_{N+1} &= \frac{1}{W_{N+1}} \sum_{i=1}^{N+1} w_i\, x_i\, y_i \qquad (3.31)\\
&= \frac{1}{W_{N+1}} \left(\sum_{i=1}^{N} w_i\, x_i\, y_i + w_{N+1}\, x_{N+1}\, y_{N+1} \right)\\
&= \frac{1}{W_{N+1}} \Big(W_N \langle xy\rangle_N + w_{N+1}\, x_{N+1}\, y_{N+1} \Big).
\end{aligned}
$$

k 번째 데이터를 하나 제거 하는 경우는 다음과 같다.

$$
\begin{aligned}
(\mathbf{A}_{N,!k})_{11} &= W_{N,!k}\Big(\langle x^2\rangle_{N,!k} - \langle x\rangle_{N,!k}^2 \Big), \qquad (3.32)\\
(\mathbf{A}_{N,!k})_{22} &= W_{N,!k}\Big(\langle y^2\rangle_{N,!k} - \langle y\rangle_{N,!k}^2 \Big), \qquad (3.33)\\
(\mathbf{A}_{N,!k})_{33} &= W_{N,!k}\Big(\langle z^2\rangle_{N,!k} - \langle z\rangle_{N,!k}^2 \Big), \qquad (3.34)\\
(\mathbf{A}_{N,!k})_{12} &= W_{N,!k}\Big(\langle xy\rangle_{N,!k} - \langle x\rangle_{N,!k} \langle y\rangle_{N,!k} \Big), \qquad (3.35)\\
(\mathbf{A}_{N,!k})_{13} &= W_{N,!k}\Big(\langle xz\rangle_{N,!k} - \langle x\rangle_{N,!k} \langle z\rangle_{N,!k} \Big), \qquad (3.36)\\
(\mathbf{A}_{N,!k})_{23} &= W_{N,!k}\Big(\langle yz\rangle_{N,!k} - \langle y\rangle_{N,!k} \langle z\rangle_{N,!k} \Big). \qquad (3.37)
\end{aligned}
$$

위에서와 같이 기댓값은 아래와 같이 구할 수 있다.

$$
\begin{aligned}
\langle x\rangle_{N,!k} &= \frac{1}{W_{N,!k}} \sum_{i=1,i\neq k}^{N} w_i\, x_i \qquad (3.38)\\
&= \frac{1}{W_{N,!k}} \left(\sum_{i=1}^{N} w_i\, x_i - w_k\, x_k \right)\\
&= \frac{1}{W_{N,!k}} \Big(W_N \langle x\rangle_N - w_k\, x_k \Big).
\end{aligned}
$$

$$
\begin{aligned}
\langle x^2\rangle_{N,!k} &= \frac{1}{W_{N,!k}} \sum_{i=1,i\neq k}^{N} w_i\, x_i^2 \qquad (3.39)\\
&= \frac{1}{W_{N,!k}} \left(\sum_{i=1}^{N} w_i\, x_i^2 - w_k\, x_k^2 \right)\\
&= \frac{1}{W_{N,!k}} \Big(W_N \langle x^2\rangle_N - w_k\, x_k^2 \Big).
\end{aligned}
$$

$$
\begin{aligned}
\langle xy\rangle_{N,!k} &= \frac{1}{W_{N,!k}} \sum_{i=1,i\neq k}^{N} w_i\, x_i\, y_i \qquad (3.40)\\
&= \frac{1}{W_{N,!k}} \left(\sum_{i=1}^{N} w_i\, x_i\, y_i - w_k\, x_k\, y_k \right)\\
&= \frac{1}{W_{N,!k}} \Big(W_N \langle xy\rangle_N - w_k\, x_k\, y_k \Big).
\end{aligned}
$$

3.3 직선 피팅

점(x_0, y_0, z_0)을 지나가고 (a, b, c)의 방향 벡터를 가지는 직선

$$
\begin{aligned}
x &= x_0 + at \\
y &= y_0 + bt \\
z &= z_0 + ct
\end{aligned}
$$

과 점(x_i, y_i, z_i) 사이의 수직 거리는 다음과 같다.

$$
D_i(x_0, y_0, z_0, a, b, c) = \begin{pmatrix} & (c(y_i - y_0) - b(z_i - z_0))^2 \\ + & (a(z_i - z_0) - c(x_i - x_0))^2 \\ + & (b(x_i - x_0) - a(y_i - y_0))^2 \end{pmatrix}^{1/2} \tag{3.41}
$$

우리는 점과 직선사이 거리 제곱의 합을 최소화 할 것이므로 S는 다음과 같이 쓸 수 있다.

$$
S(x_0, y_0, z_0, a, b, c) = w_i \sum_{i=1}^{N} \begin{pmatrix} & (c(y_i - y_0) - b(z_i - z_0))^2 \\ + & (a(z_i - z_0) - c(x_i - x_0))^2 \\ + & (b(x_i - x_0) - a(y_i - y_0))^2 \end{pmatrix} \tag{3.42}
$$

이제 S를 x_0, y_0, z_0에 대해서 각각 최소화 하면 ($\partial S/\partial x_0 = 0$, $\partial S/\partial y_0 = 0$, $\partial S/\partial z_0 = 0$) 다음식을 얻을 수 있다.

$$
\frac{x_0 - \langle x \rangle_N}{a} = \frac{y_0 - \langle y \rangle_N}{b} = \frac{z_0 - \langle z \rangle_N}{c} \tag{3.43}
$$

여기서 $\langle x \rangle_N, \langle y \rangle_N, \langle z \rangle_N$는 식 (3.4)와 같다. 위 식은 (x_0, y_0, z_0)가 $(\langle x \rangle_N, \langle y \rangle_N, \langle z \rangle_N)$일때 만족한다. 따라서 우리가 구하고자 하는 직선은 데이터의 중심점을 지나간다.

데이터 점을 R_i, 데이터의 중심점을 C, 우리가 구하고자 하는 직선을 L, 그 직선에 수직하고 점 C를 포함하는 평면을 P, 그리고 $D(\alpha, \beta)$를 α와 β 사이의 거리라고 할때, 피타고라스의 정리를 이용하면 다음과 같이 쓸 수 있다.

$$
\sum_{i=1}^{N} D(R_i, L)^2 = \sum_{i=1}^{N} D(R_i, C)^2 - \sum_{i=1}^{N} D(R_i, P)^2 \tag{3.44}
$$

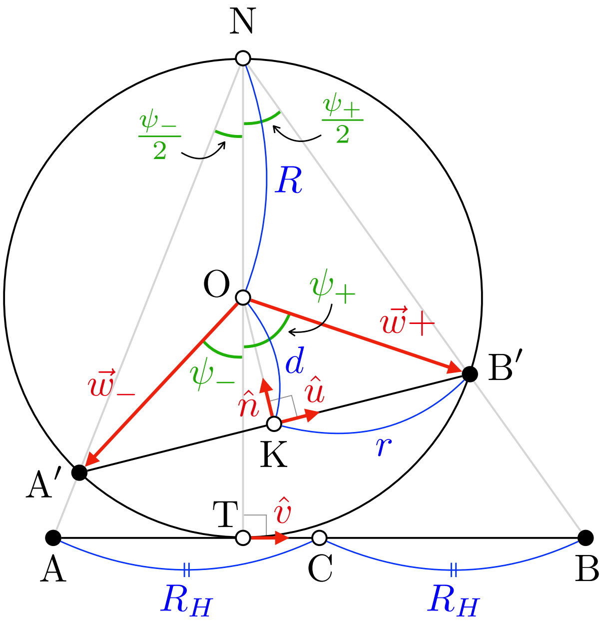
위의 $\sum D(R_i, C)^2$는 상수임을 알 수 있다. 즉 $S(= \sum D(R_i, L)^2)$는 $\sum D(R_i, P)^2$를 최대화 하므로써 구할 수 있다. 이때 $\sum D(R_i, P)^2$는 3.1절의 S(식 (3.2), (3.9))와 같으므로 행렬 $\mathbf{A}$(식 (3.19))의 가장 큰 고윳값을 찾는 문제로 귀결된다.

제 4 장

수직 거리 회귀와 Stereographic projection을 이용한 원 피팅

4.1 Stereographic projection

4.2 원 파라미터



$$n_x x + n_y y + n_z z = k \tag{4.1}$$

$$\tag{4.2}$$

$$
\begin{aligned}
k &= \hat{n} \cdot \langle \mathbf{x} \rangle_{\mathbf{M}} & (4.3) \\
d &= |k - \hat{n} \cdot \mathbf{x_{RS}}| & (4.4)
\end{aligned}
$$

$$
\begin{aligned}
\hat{u} &= \frac{l_y \, \vec{l} + m_y \, \vec{m}}{\sqrt{l_y^2 + m_y^2}} & (4.5) \\
\hat{v} &= \frac{u_x^2 \, \hat{x} + u_z^2 \, \hat{z}}{\sqrt{u_x^2 + u_z^2}} & (4.6) \\
\vec{w}_\pm &= -d\hat{n} \pm \sqrt{R_{RS}^2 - d^2} \ \hat{u} & (4.7)
\end{aligned}
$$

$$
\begin{aligned}
\psi_\pm &= \arctan \frac{|\hat{v} \cdot w_\pm|}{w_{\pm,y}} & (4.8) \\
R_\pm &= R_{RS} \left(\tan \frac{\psi_+}{2} \pm \tan \frac{\psi_-}{2} \right) & (4.9)
\end{aligned}
$$

$$
\begin{aligned}
R_H &= R_+ & (4.10) \\
\mathbf{x_H} &= R_- \hat{v} + \mathbf{x_{RS}} & (4.11)
\end{aligned}
$$

참고 문헌

[1] The Math Forum - Line of Best Fit For Points in Three Dimensional Space (http://mathforum.org/library/drmath/view/69103.html).

[2] The Math Forum - Orthogonal Distance Regression Planes (http://mathforum.org/library/drmath/view/63765.html).

[3] Singular value decomposition (https://en.wikipedia.org/wiki/Singular_value_decomposition).