Derivative of the first order SVD

Randolf Scholz

July 7, 2023

Consider computing the first order SVD expansion. By the Eckart–Young–Mirsky theorem, this is equivalent to solving

$$\underset{\sigma,\,u,\,v}{\text{minimize}}\ \tfrac{1}{2}\|A - \sigma u v^\top\|_F^2 \quad \text{s.t.} \quad \|u\| = 1 \quad \text{and} \quad \|v\| = 1 \quad \text{and} \quad \sigma \geq 0$$

Equivalently this may be formalized as

$$\sigma = \max_{u,\,v} u^\top A v \quad \text{s.t.} \quad \|u\| = 1 \quad \text{and} \quad \|v\| = 1$$

Which is a non-convex quadratically constrained quadratic program (QCQP)

$$\sigma = \max_{u,v} \frac{1}{2} \begin{bmatrix} u \\ v \end{bmatrix}^\top \cdot \begin{bmatrix} \mathbf{0}_{m\times m} & A \\ A^\top & \mathbf{0}_{n\times n} \end{bmatrix} \cdot \begin{bmatrix} u \\ v \end{bmatrix} \ \text{s.t.} \ \begin{matrix} \begin{bmatrix} u \\ v \end{bmatrix}^\top \cdot \begin{bmatrix} \mathbb{I}_m & \mathbf{0}_{m\times n} \\ \mathbf{0}_{n\times m} & \mathbf{0}_{n\times n} \end{bmatrix} \cdot \begin{bmatrix} u \\ v \end{bmatrix} = 1 \\ \begin{bmatrix} u \\ v \end{bmatrix}^\top \cdot \begin{bmatrix} \mathbf{0}_{m\times m} & \mathbf{0}_{m\times n} \\ \mathbf{0}_{n\times m} & \mathbb{I}_n \end{bmatrix} \cdot \begin{bmatrix} u \\ v \end{bmatrix} = 1 \end{matrix}$$

The Jacobian and Lagrangian The derivative of the objective function is

$$\mathbf{J}_f(A, \left[\begin{smallmatrix} \sigma \\ u \\ v \end{smallmatrix}\right]) = \begin{bmatrix} & \sigma - u^\top A v \\ A - \sigma u v^\top & \sigma^2 u - \sigma A v \\ & \sigma^2 v - \sigma A^\top u \end{bmatrix} \implies \mathbf{H}_f(\left[\begin{smallmatrix} \sigma \\ u \\ v \end{smallmatrix}\right]) = \begin{bmatrix} 1 & 2\sigma u - A v & 2\sigma v - A^\top u \\ -Av & \sigma^2 \mathbb{I}_m & -\sigma A \\ -A^\top u & -\sigma A^\top & \sigma^2 \mathbb{I}_n \end{bmatrix}$$

Consider the function

$$f(A, \left[\begin{smallmatrix} \sigma \\ u \\ v \end{smallmatrix}\right]) = \begin{pmatrix} \sigma - u^\top A v \\ \sigma^2 u - \sigma A v \\ \sigma^2 v - \sigma A^\top u \end{pmatrix} \equiv \mathbf{0} \implies \mathbf{J}_f(A, \left[\begin{smallmatrix} \sigma \\ u \\ v \end{smallmatrix}\right]) = \left[\begin{array}{c|ccc} -\xi v u^\top & 1 & 2\sigma u - A v & 2\sigma v - A^\top u \\ -\sigma v \phi^\top & -Av & \sigma^2 \mathbb{I}_m & -\sigma A \\ -\sigma u \psi^\top & -A^\top u & -\sigma A^\top & \sigma^2 \mathbb{I}_n \end{array}\right]$$

Thus, gradient descent schema is

$$\begin{aligned} \sigma' &= \sigma - \eta_\sigma(\sigma - u^\top A v) \\ u' &= u - \eta_u(\sigma^2 u - \sigma A v) \\ v' &= v - \eta_v(\sigma^2 v - \sigma A^\top u) \end{aligned}$$

And the newton step with diagonal approximation of the hessian:

$$\begin{aligned}
\sigma' &= \sigma - 1(\sigma - u^\top Av) &&= u^\top Av \\
u' &= u - \tfrac{1}{\sigma^2}(\sigma^2 u - \sigma Av) &&= \tfrac{1}{\sigma}Av \\
v' &= v - \tfrac{1}{\sigma^2}(\sigma^2 v - \sigma A^\top u) &&= \tfrac{1}{\sigma}A^\top u
\end{aligned}$$

1 Analysis of the backward

At the equilibrium point, we have:

$$\sigma = u^\top Av \qquad Av = \sigma u \qquad A^\top u = \sigma v \qquad u^\top u = 1 \qquad v^\top v = 1$$

Note that this states that σ is an eigenvalue:

$$\begin{bmatrix} 0 & A \\ A^\top & 0 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \sigma \begin{bmatrix} u \\ v \end{bmatrix}$$

In particular, Rayleigh iteration could be useful. from this we can derive

$$\Delta\sigma = \Delta u^\top Av + u^\top \Delta A v + u^\top A\Delta v = \Delta u^\top u + u^\top \Delta A v + v^\top \Delta v = u^\top \Delta A v$$

Where in the last step we used $\Delta u \perp u$ and $\Delta v \perp v$, which follows from the side condition. Further we have:

$$\begin{aligned}
\Delta\sigma u + \sigma\Delta u &= \Delta A v + A\Delta v \\
\Delta\sigma v + \sigma\Delta v &= \Delta A^\top u + A^\top \Delta u
\end{aligned} \iff \underbrace{\begin{bmatrix} \sigma\mathbb{I}_m & -A \\ -A^\top & \sigma\mathbb{I}_n \end{bmatrix}}_{=:K} \cdot \begin{bmatrix} \Delta u \\ \Delta v \end{bmatrix} = \begin{bmatrix} \Delta A v - \Delta\sigma u \\ \Delta A^\top u - \Delta\sigma v \end{bmatrix}$$

which allows us to express Δu and Δv in terms of ΔA. The constraints yield

$$\begin{aligned}
u^\top \Delta u + \Delta u^\top u = 0 &\iff u \perp \Delta u \\
v^\top \Delta v + \Delta v^\top v = 0 &\iff v \perp \Delta v
\end{aligned}$$

We can augment the original system with these:

$$\underbrace{\begin{bmatrix} \sigma\mathbb{I}_m & -A \\ -A^\top & \sigma\mathbb{I}_n \\ u^\top & \mathbf{0}_n^\top \\ \mathbf{0}_m^\top & v^\top \end{bmatrix}}_{=:\widetilde{K}} \cdot \begin{bmatrix} \Delta u \\ \Delta v \end{bmatrix} = \underbrace{\begin{bmatrix} \Delta A v - \Delta\sigma u \\ \Delta A^\top u - \Delta\sigma v \\ 0 \\ 0 \end{bmatrix}}_{=:\widetilde{c}}$$

2 VJP with modified K matrix

$$\begin{aligned}
\left\langle \begin{bmatrix}\phi\\ \psi\end{bmatrix} \;\middle|\; \begin{bmatrix}\Delta u\\ \Delta v\end{bmatrix}\right\rangle &= \left\langle \begin{bmatrix}\phi\\ \psi\end{bmatrix} \;\middle|\; \widetilde{K}^{-1}\widetilde{c}\right\rangle \\
&= \left\langle \widetilde{K}^{-\top}\begin{bmatrix}\phi\\ \psi\end{bmatrix} \;\middle|\; \widetilde{c}\right\rangle \\
&= \left\langle \begin{bmatrix}\sigma\mathbb{I}_m & -A & u & \mathbf{0}_m\\ -A^\top & \sigma\mathbb{I}_n & \mathbf{0}_n & v\end{bmatrix}^{-1} \begin{bmatrix}\phi\\ \psi\end{bmatrix} \;\middle|\; \left[\begin{smallmatrix}\Delta Av-\Delta\sigma u\\ \Delta A^\top u-\Delta\sigma v\\ 0\\ 0\end{smallmatrix}\right]\right\rangle \\
&= \left\langle \begin{bmatrix}\sigma\mathbb{I}_m & -A & u & \mathbf{0}_m\\ -A^\top & \sigma\mathbb{I}_n & \mathbf{0}_n & v\end{bmatrix} \left[\begin{smallmatrix}p\\ q\\ \lambda\\ \mu\end{smallmatrix}\right] = \begin{bmatrix}\phi\\ \psi\end{bmatrix} \;\middle|\; \left[\begin{smallmatrix}\Delta Av-\Delta\sigma u\\ \Delta A^\top u-\Delta\sigma v\\ 0\\ 0\end{smallmatrix}\right]\right\rangle
\end{aligned}$$

2.1 Augmented block inversion

NOTE: Tested this and the issue is that it vastly increases the condition number!

We use the technique Column-wise partitioning in over-determined least squares.

$$\begin{bmatrix}A & B\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix} = d \impliedby x = \begin{bmatrix}A & B\end{bmatrix}^+ d = \begin{bmatrix}(P_B^\perp A)^+\\ (P_A^\perp B)^+\end{bmatrix} d$$

In particular, in our case this means that the relevant part of the solution is

$$\begin{bmatrix}p\\ q\end{bmatrix} = (P_B^\perp K)^+ \begin{bmatrix}\phi\\ \psi\end{bmatrix}$$

Here

$$\begin{aligned}
P_B^\perp &= \mathbb{I} - BB^+ \\
&= \mathbb{I} - B(B^T B)^{-1}B^\top \\
&= \mathbb{I} - \begin{bmatrix}u & 0\\ 0 & v\end{bmatrix} \left(\begin{bmatrix}u^\top & 0^\top\\ 0^\top & v^\top\end{bmatrix}\begin{bmatrix}u & 0\\ 0 & v\end{bmatrix}\right)^{-1} \begin{bmatrix}u^\top & 0^\top\\ 0^\top & v^\top\end{bmatrix} \\
&= \mathbb{I} - \begin{bmatrix}u & 0\\ 0 & v\end{bmatrix} \left(\begin{bmatrix}u^\top & 0^\top\\ 0^\top & v^\top\end{bmatrix}\begin{bmatrix}u & 0\\ 0 & v\end{bmatrix}\right)^{-1} \begin{bmatrix}u^\top & 0^\top\\ 0^\top & v^\top\end{bmatrix} \\
&= \mathbb{I} - \begin{bmatrix}u & 0\\ 0 & v\end{bmatrix} \begin{bmatrix}1/\|u\|^2 & 0\\ 0 & 1/\|v\|^2\end{bmatrix} \begin{bmatrix}u^\top & 0^\top\\ 0^\top & v^\top\end{bmatrix} \\
&= \mathbb{I} - \begin{bmatrix}u & 0\\ 0 & v\end{bmatrix} \begin{bmatrix}1/\|u\|^2 & 0\\ 0 & 1/\|v\|^2\end{bmatrix} \begin{bmatrix}u^\top & 0^\top\\ 0^\top & v^\top\end{bmatrix} \\
&= \begin{bmatrix}\mathbb{I}_m - uu^\top & \mathbf{0}_{m\times n}\\ \mathbf{0}_{n\times m} & \mathbb{I}_n - vv^\top\end{bmatrix}
\end{aligned}$$

So

$$
\begin{aligned}
P_B^{\perp} K &= \begin{bmatrix} \mathbb{I}_m - uu^\top & \mathbf{0}_{m\times n} \\ \mathbf{0}_{n\times m} & \mathbb{I}_n - vv^\top \end{bmatrix} \begin{bmatrix} \sigma\mathbb{I}_m & -A \\ -A^\top & \sigma\mathbb{I}_n \end{bmatrix} \\
&= \begin{bmatrix} \sigma(\mathbb{I}_m - uu^\top) & -A + \sigma uv^\top \\ -A^\top + \sigma vu^\top & \sigma(\mathbb{I}_n - vv^\top) \end{bmatrix} \\
&= K - \sigma \begin{bmatrix} uu^\top & -uv^\top \\ -vu^\top & \sigma vv^\top \end{bmatrix} \\
&= K - \sigma zz^\top \qquad z = \left[\begin{smallmatrix} u \\ -v \end{smallmatrix}\right]
\end{aligned}
$$

In particular, we see that effectively this is a low rank update of the original matrix! We can use the inversion formula for 2×2 block matrices, combined with the inverse of rank-1 update formulas:

$$\beta := 1 - \sigma z^\top K^+ z = ?[\text{.. proof ...}] = 0$$

Also $z \in \mathrm{Im}(K)$, so, in particular, the case (vi) of the paper Generalized Inversion of Modified Matrices holds:

$$(A + cd^T)^+ = A^+ - kk^+A^+ - A^+h^+h + (k^+A^+h^+)kh \quad \text{with} \quad k = A^+c,\ h = d^\top A^+$$

Assuming A is symmetric, $c = d = x$ and noting that $v^+ = \frac{1}{\|v\|^2}v^\top$ we can simplify since

$$
\begin{aligned}
k &= h^\top = A^+x & k^+ &= \frac{1}{\|A^+x\|^2}x^\top A^+ \\
h &= k^\top = x^\top A^+ & h^+ &= \frac{1}{\|A^+x\|^2}A^+x
\end{aligned}
$$

$$(A + xx^T)^+ = A^+ - \frac{A^+xx^\top A^+A^+}{\|A^+x\|^2} - \frac{A^+A^+xx^\top A^+}{\|A^+x\|^2} + \frac{(x^\top(A^+)^3x)A^+xx^\top A^+}{\|A^+x\|^4}$$

With an additional scalar γ:

$$(A + \gamma xx^T)^+ = A^+ - \frac{A^+xx^\top A^+A^+}{\|A^+x\|^2} - \frac{A^+A^+xx^\top A^+}{\|A^+x\|^2} + \frac{(x^\top(A^+)^3x)A^+xx^\top A^+}{\|A^+x\|^4}$$

Now, in our case x happens to be an eigenvector: $Kz = 2\sigma z$, $K^+z = \frac{1}{2\sigma}z$, $\|z\|^2 = 2$, hence $\|K^+z\|^2 = \frac{1}{2\sigma^2}$ and $K^+zz^\top K^+ = \frac{1}{4\sigma^2}zz^\top$.

$$
\begin{aligned}
(K - \sigma zz^T)^+ &= K^+ - \tfrac{1/(2\sigma)^3}{1/2\sigma^2}zz^\top - \tfrac{1/(2\sigma)^3}{1/2\sigma^2}zz^\top + \tfrac{2/(2\sigma)^3}{1/2\sigma^2}\tfrac{1/(2\sigma)^2}{1/2\sigma^2}zz^\top \\
&= K^+ - \tfrac{1}{4\sigma}zz^\top - \tfrac{1}{4\sigma}zz^\top + \tfrac{1}{4\sigma}zz^\top \\
&= K^+ - \tfrac{1}{4\sigma}zz^\top
\end{aligned}
$$

$$\implies (A + xx^T)^+ =$$

Note that $v^+ = \frac{1}{\|v\|^2}v^\top$.

3 The VJP

The last equation allows us to compute the VJP at ease:

$$\begin{aligned}
\left\langle \begin{bmatrix}\phi\\ \psi\end{bmatrix} \,\middle|\, \begin{bmatrix}\Delta u\\ \Delta v\end{bmatrix} \right\rangle &= \left\langle \begin{bmatrix}\phi\\ \psi\end{bmatrix} \,\middle|\, K^{-1} \begin{bmatrix}\Delta A v - \Delta\sigma u\\ \Delta A^\top u - \Delta\sigma v\end{bmatrix} \right\rangle \\
&= \left\langle K^{-\top} \begin{bmatrix}\phi\\ \psi\end{bmatrix} \,\middle|\, \begin{bmatrix}\Delta A v - \Delta\sigma u\\ \Delta A^\top u - \Delta\sigma v\end{bmatrix} \right\rangle \\
&= \left\langle \begin{bmatrix}\tilde\phi\\ \tilde\psi\end{bmatrix} \,\middle|\, \begin{bmatrix}\Delta A v - \Delta\sigma u\\ \Delta A^\top u - \Delta\sigma v\end{bmatrix} \right\rangle
\end{aligned}$$

Now, we compute the terms individually:

$$\begin{aligned}
\langle \tilde\phi \mid \Delta A v - \Delta\sigma u\rangle &= \langle \tilde\phi v^\top \mid \Delta A\rangle - \langle u^\top \tilde\phi \mid \Delta\sigma\rangle \\
&= \langle \tilde\phi v^\top \mid \Delta A\rangle - \langle u^\top \tilde\phi \mid u^\top \Delta A v\rangle \\
&= \langle (\mathbb{I}_m - uu^\top)\tilde\phi v^\top \mid \Delta A\rangle
\end{aligned}$$

And for the second term we get

$$\begin{aligned}
\langle \tilde\psi \mid \Delta A^\top u - \Delta\sigma v\rangle &= \langle \tilde\psi u^\top \mid \Delta A^\top\rangle - \langle v^\top \tilde\psi \mid \Delta\sigma\rangle \\
&= \langle u\tilde\psi^\top \mid \Delta A\rangle - \langle \tilde\psi^\top v \mid u^\top \Delta A v\rangle \\
&= \langle u\tilde\psi(\mathbb{I}_n - vv^\top) \mid \Delta A\rangle
\end{aligned}$$

Using the formula for inverting a 2×2 block-matrix, we can give an explicit solution to $K^{-\top}\left[\begin{smallmatrix}\phi\\ \psi\end{smallmatrix}\right]$:

$$\begin{aligned}
K^{-1} = \begin{bmatrix}\sigma\mathbb{I}_m & -A\\ -A^\top & \sigma\mathbb{I}_n\end{bmatrix}^{-1} &= \begin{bmatrix}(\sigma\mathbb{I}_m - \frac{1}{\sigma}AA^\top)^{-1} & \mathbf{0}_{m\times n}\\ \mathbf{0}_{n\times m} & (\sigma\mathbb{I}_n - \frac{1}{\sigma}A^\top A)^{-1}\end{bmatrix} \cdot \begin{bmatrix}\mathbb{I}_m & \frac{1}{\sigma}A\\ \frac{1}{\sigma}A^\top & \mathbb{I}_n\end{bmatrix} \\
&= \begin{bmatrix}\sigma(\sigma^2\mathbb{I}_m - AA^\top)^{-1} & (\sigma^2\mathbb{I}_m - AA^\top)^{-1}A\\ (\sigma^2\mathbb{I}_n - A^\top A)^{-1}A^\top & \sigma(\sigma^2\mathbb{I}_n - A^\top A)^{-1}\end{bmatrix}
\end{aligned}$$

And we see it's basically projection operators with respect to the image/kernel of $\tilde A = \frac{1}{\sigma}A$. In summary, we obtain the following formula for the VJP:

$$K\begin{bmatrix}p\\ q\end{bmatrix} = \begin{bmatrix}\phi\\ \psi\end{bmatrix} \iff \begin{bmatrix}p\\ q\end{bmatrix} = \begin{bmatrix}\sigma(\sigma^2\mathbb{I}_m - AA^\top)^{-1} & (\sigma^2\mathbb{I}_m - AA^\top)^{-1}A\\ (\sigma^2\mathbb{I}_n - A^\top A)^{-1}A^\top & \sigma(\sigma^2\mathbb{I}_n - A^\top A)^{-1}\end{bmatrix}\begin{bmatrix}\phi\\ \psi\end{bmatrix}$$

In particular, we can find the solution by solving 4 smaller linear systems:

$$\begin{aligned}
\sigma(\sigma^2\mathbb{I}_m - AA^\top)^{-1}\phi &= x & \qquad (\sigma^2\mathbb{I}_m - AA^\top)^{-1}A\psi &= y\\
(\sigma^2\mathbb{I}_n - A^\top A)^{-1}A^\top\phi &= w & \qquad \sigma(\sigma^2\mathbb{I}_n - A^\top A)^{-1}\psi &= z
\end{aligned}$$

Or, equivalently:

$$\begin{aligned}
(\sigma^2\mathbb{I}_m - AA^\top)x &= \sigma\phi & \qquad (\sigma^2\mathbb{I}_m - AA^\top)y &= A\psi\\
(\sigma^2\mathbb{I}_n - A^\top A)w &= A^\top\phi & \qquad (\sigma^2\mathbb{I}_n - A^\top A)z &= \sigma\psi
\end{aligned}$$

Note how this shows that the off-diagonal entries are solutions to regularized least squares problems! However, we really do not want to compute the matrices $AA^\top$ and $A^\top A$ since this leads to numerical stability (squared condition number!) To circumvent this issue, we do a reformulation

$$\begin{aligned}
(\sigma^2\mathbb{I}_m - AA^\top)y = A\psi \quad &\iff y = \operatorname*{argmin}_y \| - A^\top y - \psi\|_2^2 - \sigma^2\|y\|_2^2 \\
&\iff y = \operatorname*{argmin}_y \left\| \begin{bmatrix} A^\top \\ \sigma^2\mathbb{I}_m \end{bmatrix} y - \begin{bmatrix} -\psi \\ \mathbf{0}_m \end{bmatrix} \right\|_2^2 \\
(\sigma^2\mathbb{I}_n - A^\top A)w = A^\top\phi &\iff w = \operatorname*{argmin}_w \|Aw + \phi\|_2^2 - \sigma^2\|w\|_2^2 \\
&\iff w = \operatorname*{argmin}_w \left\| \begin{bmatrix} A \\ \sigma^2\mathbb{I}_n \end{bmatrix} w - \begin{bmatrix} -\phi \\ \mathbf{0}_n \end{bmatrix} \right\|_2^2
\end{aligned}$$

Remark 1 (When is Ridge Regression unconstrained?). Consider the problem

$$\beta^* = \operatorname*{argmin}_\beta \|X\beta - y\|^2 + \lambda\|\beta\|^2$$

Question: When is there an unconstrained solution? The solution satisfies the normal equation

$$(X^T X + \lambda\mathbb{I})\beta = X^\top y$$

If $\lambda > 0$, then $(X^T X + \lambda\mathbb{I})$ is positive definite and hence invertible. If $\lambda < 0$, then $(X^T X + \lambda\mathbb{I})$ is singular whenever λ is an eigenvalue of $X^T X$. In particular, the 4 systems listed before are all ill-conditioned! The central issue is that the constraint is missing! $\|u\|^2 = 1$ and $\|v\|^2 = 1$ translate to $u \perp \Delta u$ and $v \perp \Delta v$. Since u, v are singular vectors, this means we avoid the singular subspace when solving these equations!

What we should do is use **Riemannian Optimization**.

3.1 What happens if ϕ or ψ are zero?

In this case we want to fast track the calculation, meaning skip half of the necessary inversions. Looking at the equations we find that if $\phi = 0$ then $x = 0$ and $w = 0$, and if $\psi = 0$ then $y = 0$ and $z = 0$. This suggests that backward substitution is better than forward substitution, since it allows decoupling of the two gradient contributions.

3.2 Via Forward Substitution

Now, the diagonal entries we have a problem: the RHS lacks the A matrix. Thus, we solve in two steps instead:

$$\begin{aligned}
A\mu = \sigma\phi &\implies x = \operatorname*{argmin}_x \left\| \begin{bmatrix} A^\top \\ \sigma^2\mathbb{I}_m \end{bmatrix} x - \begin{bmatrix} -\mu \\ \mathbf{0}_m \end{bmatrix} \right\|_2^2 \\
A^\top\nu = \sigma\psi &\implies z = \operatorname*{argmin}_z \left\| \begin{bmatrix} A \\ \sigma^2\mathbb{I}_n \end{bmatrix} z - \begin{bmatrix} -\nu \\ \mathbf{0}_n \end{bmatrix} \right\|_2^2
\end{aligned}$$

We can optimize further by performing a simultaneous solve:

$$
\begin{aligned}
\begin{bmatrix} x, & y \end{bmatrix} &= \operatorname*{argmin}_{x,y} \left\| \begin{bmatrix} A^\top \\ \sigma^2 \mathbb{I}_m \end{bmatrix} \begin{bmatrix} x, & y \end{bmatrix} - \begin{bmatrix} -\mu & -\psi \\ \mathbf{0}_m & \mathbf{0}_m \end{bmatrix} \right\|_2^2 & \mu &= \operatorname*{argmin}_{\mu} \|A\mu - \sigma\phi\|_2^2 \\
\begin{bmatrix} w, & z \end{bmatrix} &= \operatorname*{argmin}_{w,z} \left\| \begin{bmatrix} A \\ \sigma^2 \mathbb{I}_n \end{bmatrix} \begin{bmatrix} w, & z \end{bmatrix} - \begin{bmatrix} -\phi & -\nu \\ \mathbf{0}_n & \mathbf{0}_n \end{bmatrix} \right\|_2^2 & \nu &= \operatorname*{argmin}_{\nu} \|A^\top\nu - \sigma\psi\|_2^2
\end{aligned}
$$

3.3 Via Backward Substitution

We need to introduce an additional modification:

If $A\mu = \sigma\phi$ not solveable, we instead can multiply the equation by $A^\top$ to obtain:

$$
\begin{aligned}
(\sigma^2\mathbb{I}_m - AA^\top)x = \sigma\phi \quad &\Longrightarrow \quad (\sigma^2\mathbb{I}_n - A^\top A)\mu = \sigma A^\top\phi \quad & A^\top x = \mu \\
(\sigma^2\mathbb{I}_n - A^\top A)z = \sigma\psi \quad &\Longrightarrow \quad (\sigma^2\mathbb{I}_n - A^\top A)A\nu = \sigma A\psi \quad & Az = \nu
\end{aligned}
$$

So:

$$
\begin{aligned}
\mu &= \operatorname*{argmin}_{\mu} \left\| \begin{bmatrix} A \\ \sigma^2 \mathbb{I}_n \end{bmatrix} \mu - \begin{bmatrix} -\sigma\phi \\ \mathbf{0}_n \end{bmatrix} \right\|_2^2 & A^\top x &= \mu \\
\nu &= \operatorname*{argmin}_{\mu} \left\| \begin{bmatrix} A^\top \\ \sigma^2 \mathbb{I}_m \end{bmatrix} \nu - \begin{bmatrix} -\sigma\psi \\ \mathbf{0}_m \end{bmatrix} \right\|_2^2 & Az &= \nu
\end{aligned}
$$

So

$$
\begin{aligned}
\begin{bmatrix} \mu & w \end{bmatrix} &= \begin{bmatrix} A \\ \sigma^2 \mathbb{I}_n \end{bmatrix} \begin{bmatrix} -\sigma\phi & -\phi \\ \mathbf{0}_n & \mathbf{0}_n \end{bmatrix} \\
\begin{bmatrix} y & \nu \end{bmatrix} &= \begin{bmatrix} A^\top \\ \sigma^2 \mathbb{I}_m \end{bmatrix} \begin{bmatrix} -\psi & -\sigma\psi \\ \mathbf{0}_m & \mathbf{0}_m \end{bmatrix}
\end{aligned}
$$

In principle, one could try to rephrase these as smaller problems, but for now, it's better to just stick to the bigger system. We can use the **push-through identity** to convert these into 4 linear systems:

$$
\begin{aligned}
Px &= \phi & Py &= \tilde{A}\psi \\
Qz &= \tilde{A}^\top\phi & Qw &= \psi
\end{aligned}
$$

Then $\tilde{\phi} = x + y$ and $\tilde{\psi} = z + w$, and the VJP are given by the previous equations:

$$
\begin{aligned}
\xi^\top \frac{\partial\sigma}{\partial A} &= \xi u v^\top \\
\phi^\top \frac{\partial u}{\partial A} &= (\mathbb{I}_m - uu^\top)\tilde{\phi}v^\top = (\tilde{\phi} - (u^\top\tilde{\phi})u)v^\top \\
\psi^\top \frac{\partial v}{\partial A} &= u\tilde{\psi}^\top(\mathbb{I}_n - vv^\top) = u(\tilde{\psi} - (v^\top\tilde{\psi})v)^\top
\end{aligned}
$$

4 Spectral Normalization

The VJP of spectral normalization can be computed as follows: let $g(A) = \|A\|_2$ and V be the vector in the VJP. then

$$\begin{aligned}
\nabla_A \langle V \mid \frac{A}{\|A\|_2} \rangle &= \langle V | \frac{A + \Delta A}{g(A+\Delta A)} - \frac{A}{g(A)} \rangle \\
&= \langle V | \frac{A + \Delta A}{g(A) + \nabla g(A)\Delta A} - \frac{A}{g(A)} \rangle \\
&= \langle V | \frac{(A + \Delta A)(g(A) - \nabla g(A)\Delta A)}{(g(A) + \nabla g(A)\Delta A)(g(A) - \nabla g(A)\Delta A)} - \frac{A}{g(A)} \rangle \\
&= \langle V | \frac{\Delta A g(A) - A \nabla g(A) \Delta A}{g(A)^2} \rangle \\
&= \langle \tfrac{1}{g(A)} V - \tfrac{\langle V|A\rangle}{g(A)} \nabla g(A) | \Delta A \rangle
\end{aligned}$$

$$g(A) = 1 \implies \nabla_A \langle V \mid \frac{A}{\|A\|_2} \rangle = \langle V - \langle V \mid A \rangle \nabla g(A) \mid \Delta A \rangle$$

5 Projected gradient

When using spectral normalization we want to do the following:

$$\begin{aligned}
\text{update:} \quad & A' = A - \nabla_A \mathcal{L}(\frac{A}{\|A\|_2}) \\
\text{project:} \quad & A = \frac{A'}{\|A'\|_2}
\end{aligned}$$

Moreover, we want:

- During forward, compute $\frac{A}{\|A\|_2}$ only once and then reuse this node.
- Compute $\|A\|_2$ effectively between gradient updates.
  - Avoid built-in torch algos, as they make use of full SVD algos.
- After gradient update, perform projection step. (maybe unnecessary)

NOTE: gradients are different if we include normalization!

6 New Approach: As a 2 player game

We recognize that the problem can also be consider an instance of a **bilinear program**. In particular, it is bi-convex: $f(u, v) = u^\top A v$ is convex both in u and v in isolation, but not together.

Reformulate the problem as a 2 player game:

- Player ①: $\max_{u:\|u\|=1} u^\top Av \rightsquigarrow$ Lagrangian $\mathscr{L} = u^\top Av + \lambda(u^\top u - 1)$
- Player ①: $\max_{v:\|v\|=1} u^\top Av \rightsquigarrow$ Lagrangian $\mathscr{L} = u^\top Av + \mu(v^\top v - 1)$

6.1 Excursion: equality constrained Newton method

Consider the equality constrained problem

$$\min_x f(x) \quad \text{s.t.} \quad h(x) = 0 \tag{1}$$

Where $f(x)$ is strictly convex and twice differentiable and the k-many constraints $h\colon \mathbb{R}^d \to \mathbb{R}^k$ are differentiable.

Theorem 2. *The minimizer x^* of (1) satisfies $\nabla f(x^*) = \nabla h(x^*)\boldsymbol{\lambda}$ for some $\boldsymbol{\lambda} \in \mathbb{R}^k$. In other words, the gradient of the objective function is a linear combination of the gradients of the constraints. 2In the special case $k = 1$, the gradients are* ***parallel****.*

The newton update can now be derived ass follows: We approximate the function locally by its second order Taylor expansion:

$$\begin{aligned} &\min_x f(x) \\ &\text{s.t. } h(x) = 0 \end{aligned} \quad \longrightarrow \quad \begin{aligned} &\min_{\Delta x} f(x + \Delta x) \\ &\text{s.t. } h(x + \Delta x) = 0 \end{aligned}$$

Which upon Taylor expansion becomes

$$\begin{aligned} &\min_{\Delta x} f(x) + \nabla f(x)^\top \Delta x + \tfrac{1}{2}\Delta x^\top \nabla^2 f(x) \Delta x \\ &\text{s.t. } h(x) + \nabla h(x)^\top \Delta x = 0 \end{aligned}$$

Which turns the problem in a convex quadratic optimization with linear constraint, which can be solved analytically. The Lagrangian is

$$\mathscr{L}(\Delta x, \lambda) = f(x) + \nabla f(x)^\top \Delta x + \tfrac{1}{2}\Delta x^\top \nabla^2 f(x) \Delta x + (h(x) + \nabla h(x)\Delta x)^\top \boldsymbol{\lambda}$$

So the first order KKT conditions are obtained by computing $\nabla_{\Delta x}\mathscr{L}$:

$$\begin{aligned} 0 &= \nabla^2 f(x)\Delta x + \nabla f(x) + \lambda^\top \nabla h(x) \\ 0 &= h(x) + \nabla h(x)^\top \Delta x \end{aligned} \iff \begin{bmatrix} \nabla^2 f(x) & \nabla h(x)^\top \\ \nabla h(x) & 0 \end{bmatrix} \cdot \begin{bmatrix} \Delta x \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} -\nabla f(x) \\ -h(x) \end{bmatrix}$$

6.2 Application to the bilinear game

We assume each player estimates their updated objective function with a linear approximation:

$$(u + \Delta u)^\top A(v + \Delta v) = u^\top Av + \Delta u^\top Av + u^\top A\Delta v + \Delta u^\top A\Delta v$$

And the expansions of the constraints are

$$\begin{aligned} h_1(u+\Delta u) &= \|u+\Delta u\|^2 - 1 \approx \|u\|^2 + 2\langle u \mid \Delta u\rangle - 1 \\ h_2(v+\Delta v) &= \|v+\Delta v\|^2 - 1 \approx \|v\|^2 + 2\langle v \mid \Delta v\rangle - 1 \end{aligned}$$

Hence the Lagrangians are:

$$\begin{aligned} \mathscr{L}_1(\Delta u,\, \lambda) &= u^\top Av + \Delta u^\top Av + u^\top A\Delta v + \Delta u^\top A\Delta v + \lambda(\|u\|^2 + 2\langle u \mid \Delta u\rangle - 1) \\ \mathscr{L}_2(\Delta v,\, \mu) &= u^\top Av + \Delta u^\top Av + u^\top A\Delta v + \Delta u^\top A\Delta v + \mu(\|v\|^2 + 2\langle v \mid \Delta v\rangle - 1) \end{aligned}$$

So, the first order conditions for each player are:

$$\begin{aligned} 0 &\overset{!}{=} \nabla_{\Delta u}\mathscr{L}_1(\Delta u,\, \lambda) &&= Av + A\Delta v + 2\lambda u \\ 0 &\overset{!}{=} \nabla_{\Delta v}\mathscr{L}_2(\Delta v,\, \lambda) &&= A^\top u + A^\top \Delta u + 2\lambda v \\ 0 &\overset{!}{=} h_1(u+\Delta u) &&= \|u\|^2 + 2\langle u \mid \Delta u\rangle - 1 \\ 0 &\overset{!}{=} h_1(v+\Delta v) &&= \|v\|^2 + 2\langle v \mid \Delta v\rangle - 1 \end{aligned}$$

Which gives rise to a linear system with block structure:

$$\left[\begin{array}{cc|cc} \mathbf{0} & A & 2u & 0 \\ A^\top & \mathbf{0} & 0 & 2v \\ \hline 2u^\top & 0 & 0 & 0 \\ 0 & 2v^\top & 0 & 0 \end{array}\right] \cdot \left[\begin{array}{c} \Delta u \\ \Delta v \\ \hline \lambda \\ \mu \end{array}\right] = \left[\begin{array}{c} -Av \\ -A^\top u \\ \hline 1-\|u\|^2 \\ 1-\|v\|^2 \end{array}\right] \tag{2}$$

ISSUE: Really bad conditioning!! QUESTION: Can we find a clever block-factorization of this matrix? Additionally, one can consider adding additional terms $\frac{1}{2}\eta_u\|\Delta u\|^2$ and $\frac{1}{2}\eta_v\|\Delta v\|^2$ that model the loss of trust in the approximation for large Δu and Δv. Adding these terms gives the equation

$$\left[\begin{array}{cc|cc} \eta_u\mathbb{I}_m & A & 2u & 0 \\ A^\top & \eta_v\mathbb{I}_n & 0 & 2v \\ \hline 2u^\top & 0 & 0 & 0 \\ 0 & 2v^\top & 0 & 0 \end{array}\right] \cdot \left[\begin{array}{c} \Delta u \\ \Delta v \\ \hline \lambda \\ \mu \end{array}\right] = \left[\begin{array}{c} -Av \\ -A^\top u \\ \hline 1-\|u\|^2 \\ 1-\|v\|^2 \end{array}\right] \tag{3}$$

To which we can apply **block inversion**.

$$\left[\begin{array}{cc|cc} \eta_u\mathbb{I}_m & A & 2u & 0 \\ A^\top & \eta_v\mathbb{I}_n & 0 & 2v \\ \hline 2u^\top & 0 & 0 & 0 \\ 0 & 2v^\top & 0 & 0 \end{array}\right]^{-1} = \begin{bmatrix} X & B \\ B^\top & \mathbf{0} \end{bmatrix}^{-1} = \begin{bmatrix} X^{-1} + X^{-1}BYB^\top X^{-1} & -X^{-1}BY \\ -YB^\top X^{-1} & Y \end{bmatrix}$$

Where $Y = -(B^\top X^{-1}B)^{-1}$ is the **inverse schur complement** M/X.

By the same technique, the inverse of X is

$$
\begin{aligned}
X^{-1} &= \begin{bmatrix} \eta_u \mathbb{I}_m & A \\ A^\top & \eta_v \mathbb{I}_n \end{bmatrix}^{-1} \\
&= \begin{bmatrix} (\eta_u \mathbb{I}_m + \frac{1}{\eta_v} A A^\top)^{-1} & 0 \\ 0 & (\eta_v \mathbb{I}_n + \frac{1}{\eta_u} A^\top A)^{-1} \end{bmatrix} \cdot \begin{bmatrix} \mathbb{I}_m & -\frac{1}{\eta_v} A \\ -\frac{1}{\eta_u} A^\top & \mathbb{I}_n \end{bmatrix} \\
&= \begin{bmatrix} (\eta_u \mathbb{I}_m + \frac{1}{\eta_v} A A^\top)^{-1} & -\frac{1}{\eta_v} (\eta_u \mathbb{I}_m + \frac{1}{\eta_v} A A^\top)^{-1} A \\ -\frac{1}{\eta_u} (\eta_v \mathbb{I}_n + \frac{1}{\eta_u} A^\top A)^{-1} A^\top & (\eta_v \mathbb{I}_n + \frac{1}{\eta_u} A^\top A)^{-1} \end{bmatrix}
\end{aligned}
$$

In particular:

$$
X^{-1} B = 2 \begin{bmatrix} (\eta_u \mathbb{I}_m + \frac{1}{\eta_v} A A^\top)^{-1} u & -\frac{1}{\eta_v} (\eta_u \mathbb{I}_m + \frac{1}{\eta_v} A A^\top)^{-1} A v \\ -\frac{1}{\eta_u} (\eta_v \mathbb{I}_n + \frac{1}{\eta_u} A^\top A)^{-1} A^\top u & (\eta_v \mathbb{I}_n + \frac{1}{\eta_u} A^\top A)^{-1} v \end{bmatrix}
$$

6.3 Second Order expansion of constraints

In the above the system matrix has really bad conditioning. What happens if we expand h by a quadratic term?

$$
\begin{aligned}
\mathscr{L}_1(\Delta u, \lambda) &= u^\top A v + \Delta u^\top A v + u^\top A \Delta v + \Delta u^\top A \Delta v + \lambda(\|u\|^2 + 2\langle u \mid \Delta u \rangle + \|\Delta u\|^2 - 1) \\
\mathscr{L}_2(\Delta v, \mu) &= u^\top A v + \Delta u^\top A v + u^\top A \Delta v + \Delta u^\top A \Delta v + \mu(\|v\|^2 + 2\langle v \mid \Delta v \rangle + \|\Delta u\|^2 - 1)
\end{aligned}
$$

Then, the first order conditions for each player are:

$$
\begin{aligned}
0 &\overset{!}{=} \nabla_{\Delta u} \mathscr{L}_1(\Delta u, \lambda) &&= A v + A \Delta v + 2\lambda u + 2\lambda \Delta u \\
0 &\overset{!}{=} \nabla_{\Delta v} \mathscr{L}_2(\Delta v, \lambda) &&= A^\top u + A^\top \Delta u + 2\lambda v + 2\lambda \Delta v \\
0 &\overset{!}{=} h_1(u + \Delta u) &&= \|u\|^2 + 2\langle u \mid \Delta u \rangle + \|\Delta u\|^2 - 1 \\
0 &\overset{!}{=} h_1(v + \Delta v) &&= \|v\|^2 + 2\langle v \mid \Delta v \rangle + \|\Delta u\|^2 - 1
\end{aligned}
$$

Which is no longer a linear system due to the bilinear terms.

7 Exponentiation trick

An issue with applying the Newton method for this problem is that the Newton method does not distinguish between Mimima, Saddle points and Maxima.

A trick we can use is to exponentiating the objective function:

$$
\begin{aligned}
&\max_{u,v} u^\top A v & & \max_{u,v} e^{u^\top A v} \\
&\text{s.t. } \|u\| = 1 & \longrightarrow \quad & \text{s.t. } \|u\| = 1 \\
&\qquad \|v\| = 1 & & \qquad \|v\| = 1
\end{aligned}
$$

The difference is that the second order expansion now becomes:

$$
\begin{aligned}
e^{(u+\Delta u)^\top A(v+\Delta v)} = e^{u^\top Av}(1 + \Delta u^\top Av + u^\top A\Delta v \\
+ \tfrac{1}{2}\Delta v^\top A^\top uu^\top A\Delta v + \tfrac{1}{2}\Delta uAvvA^\top \Delta u + \Delta u^\top A\Delta v)
\end{aligned}
$$

Hence the Lagrangians get the extra terms

$$
\begin{aligned}
0 &\overset{!}{=} \nabla_{\Delta u}\mathscr{L}_1(\Delta u,\, \lambda) &&= AvvA^\top \Delta u + Av + A\Delta v + 2\lambda u + 2\lambda\Delta u \\
0 &\overset{!}{=} \nabla_{\Delta v}\mathscr{L}_2(\Delta v,\, \lambda) &&= A^\top uu^\top A\Delta v + A^\top u + A^\top \Delta u + 2\lambda v + 2\lambda\Delta v \\
0 &\overset{!}{=} h_1(u + \Delta u) &&= \|u\|^2 + 2\langle u \mid \Delta u\rangle + \|\Delta u\|^2 - 1 \\
0 &\overset{!}{=} h_1(v + \Delta v) &&= \|v\|^2 + 2\langle v \mid \Delta v\rangle + \|\Delta u\|^2 - 1
\end{aligned}
$$

So the modified system is

$$
\left[\begin{array}{cc|cc}
AvvA^\top & A & 2u & 0 \\
A^\top & A^\top uu^\top A & 0 & 2v \\
\hline
2u^\top & 0 & 0 & 0 \\
0 & 2v^\top & 0 & 0
\end{array}\right]
\cdot
\left[\begin{array}{c}
\Delta u \\
\Delta v \\
\hline
\lambda \\
\mu
\end{array}\right]
=
\left[\begin{array}{c}
-Av \\
-A^\top u \\
\hline
1 - \|u\|^2 \\
1 - \|v\|^2
\end{array}\right]
\tag{4}
$$

Substituting $\tilde{u} \ = Av$ and $\tilde{v} = A^\top u$ this becomes

$$
\left[\begin{array}{cc|cc}
\tilde{u}\tilde{u}^\top & A & 2u & 0 \\
A^\top & \tilde{v}\tilde{v}^\top & 0 & 2v \\
\hline
2u^\top & 0 & 0 & 0 \\
0 & 2v^\top & 0 & 0
\end{array}\right]
\cdot
\left[\begin{array}{c}
\Delta u \\
\Delta v \\
\hline
\lambda \\
\mu
\end{array}\right]
=
\left[\begin{array}{c}
-\tilde{u} \\
-\tilde{v} \\
\hline
1 - \|u\|^2 \\
1 - \|v\|^2
\end{array}\right]
\tag{5}
$$

8 Relaxation as Second Order Cone Program

A *second order cone program* (SOCP) is defined as

$$
\min_x f^\top x \quad \text{s.t.} \quad \|A_i x + b_i\|_2 \le c_i^\top x + d_i, \quad i = 1, \ldots, m
$$