Fast method in evaluating density of multivariate Gaussian

- Given data $x \in \mathbb{R}^d$, the likelihood that it comes from a multivariate Gaussian density with mean vector $\mu \in \mathbb{R}^d$ and covariance matrix $\Sigma \in \mathbb{R}^{d \times d}$ is

$$\mathcal{N}(x; \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^d \text{det}(\Sigma)}} e^{-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)}$$

- The most expensive part to compute this is to evaluate Σ^{-1}, which has a complexity $\mathcal{O}(d^3)$.
- Moreover, when Σ is rank-deficient, i.e., there are close-to-zero eigenvalues, computing Σ^{-1} will return NAN (you cannot invert the matrix)

- Now let's make it faster and avoid the numerical issues by compute using "low-rank approximation"
- Compute eigendecomposition of

$$\Sigma = U\Lambda U^T$$

where $\Lambda = \text{diag}\{\lambda_1, \ldots, \lambda_d\}$ and the eigenvalues are ordered

$$\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_d$$

- The rank-r approximation ($r < d$) of Σ is

$$\tilde{\Sigma} = \tilde{U}\tilde{\Lambda}\tilde{U}^T$$

where $\tilde{U}$ is a d-by-r matrix formed by the first r columns of U, $\tilde{\Lambda} = \text{diag}\{\lambda_1, \ldots, \lambda_r\}$.
- Typically we will choose r such that at least $\lambda_r > 0$

- Now compute transform of data and parameters

$$\tilde{x} = \tilde{U}^T x$$
$$\tilde{\mu} = \tilde{U}^T \mu$$

https://www.cse.huji.ac.il/~csip/tirgul34.pdf

- Compute $\tilde{\Lambda}^{-1} = \text{diag}\{\lambda_1^{-1}, \ldots, \lambda_r^{-1}\}$
- Note that

$$\det(\Sigma) = \prod_{i=1}^{d} \lambda_i, \quad \det(\tilde{\Sigma}) = \prod_{i=1}^{r} \lambda_i$$

- Finally, the density calculated by replacing Σ with $\tilde{\Sigma}$ is:

$$\mathcal{N}(x; \mu, \Sigma) \approx \frac{1}{\sqrt{(2\pi)^d \prod_{i=1}^{r} \lambda_i}} \exp\left\{ -\frac{1}{2} \sum_{i=1}^{r} \frac{(\tilde{x}_i - \tilde{\mu}_i)^2}{\lambda_i} \right\}$$

  where $\tilde{x}_i$ and $\tilde{\mu}_i$ denote the ith entry of $\tilde{x}$ and $\tilde{\mu}$, respectively.
- Note: you can play with different r to have a good tradeoff between accuracy and speed

- Note that above we have used the following basic identity from linear algebra

$$\tilde{\Sigma}^{-1} = \tilde{U}\tilde{\Lambda}^{-1}\tilde{U}^T$$

and

$$\begin{aligned}
&(x-\mu)^T\Sigma^{-1}(x-\mu) \\
=&(x-\mu)^T\tilde{U}\tilde{\Lambda}^{-1}\tilde{U}^T(x-\mu) \\
=&[\tilde{U}^T(x-\mu)]^T\tilde{\Lambda}^{-1}[\tilde{U}^T(x-\mu)] \\
=&[\tilde{x}-\tilde{\mu}]^T\tilde{\Lambda}^{-1}[\tilde{x}-\tilde{\mu}] \\
=&\sum_{i=1}^{r}\frac{(\tilde{x}_i-\tilde{\mu}_i)^2}{\lambda_i}
\end{aligned}$$