Spectral Learning

- The Method of Moments for Estimation.

**Estimation:**

Given \( x_1, \ldots, x_n \overset{i.i.d.}{\sim} P_{\theta} \), a class \( \Theta \) s.t. \( \theta \in \Theta \), find \( \hat{\theta} \), an estimate of \( \theta \).

**Method of Moments:**

In Method of moments, we have equations of the form:

\[
\mathbb{E}[x] = f_1(\theta)
\]

\[
\mathbb{E}[x^2] = f_2(\theta)
\]

\[
\vdots = \text{etc.}
\]

We calculate the empirical moments: based on an iid sample

\[
\hat{f_1}(\hat{\theta})
\]

\[
\hat{f_2}(\hat{\theta})
\]

\[
\vdots = \text{etc}
\]

and solve for \( \hat{\theta} \).

**Example:** For normals, \( \Theta = (\mu, \sigma^2) \),

\[
\mathbb{E}(x) = \mu
\]

\[
\mathbb{E}(x^2) = \mu^2 + \sigma^2
\]

Solve for \( \hat{\mu} = \bar{x} \)

\[
\hat{\mu}^2 + \hat{\sigma}^2 = \hat{\mathbb{E}}(x^2)
\]

**Question:** How to do this accurately for models with a large number of parameters?

**Answer:** In some cases, through tensor decomposition.
Matrix and Tensor Decompositions:

Symmetric matrix $A_{n \times n}$

Eigen decomposition:

$$A = \sum_{i=1}^{n} \lambda_i u_i u_i^T$$

$u_i$ = eigenvectors

$u_i \perp u_j$, $i \neq j$

$A$ = symmetric matrix $A_{m \times n}$, $m \leq n$

$$A = \sum_{i=1}^{n} \lambda_i u_i u_i^T$$

$\lambda_i \geq 0$

$k_{\text{best}}$ such that

$$\sum_{i=1}^{k} \lambda_i u_i u_i^T \approx A_k$$

$\text{rank } k$ approx. to $A$.

(minimizes $\|A - A_k\|_F$ and $\|A - A_k\|_F$).

Tensors:

$T_{m_1 \times n_2 \times n_3}$

* Tensor notation: $\mathcal{A} \otimes \mathcal{B} \otimes \mathcal{C} = \mathcal{M} \times \mathcal{N} \times \mathcal{P}$ tensor $T$ with

$$T_{ijk} = a_{ij} b_{jk} c_{ik}$$

* Kronecker product:

$$\mathcal{A} \otimes \mathcal{B} \otimes \mathcal{C} \rightarrow \mathcal{M} \times \mathcal{N} \times \mathcal{P}$$

such that:

$$T_{ij k} = \sum_{i' j' k'} T_{i' j' k'} A_{i' j'} C_{k'}$$

Math world: $\sum$ indices needed for $\sum$ 2 times.

With a formula:

$$\sum_{i' j' k'} T_{i' j' k'} A_{i' j'} C_{k'}$$

$T \left( \mathcal{A}, \mathcal{B}, \mathcal{C} \right) = \sum_{i' j' k'} T_{i' j' k'} A_{i' j'} C_{k'}$$

$\mathcal{A} \otimes \mathcal{B} \otimes \mathcal{C}$
How to compute decompositions?

Matrix decomposition: Power Method (for symmetric)

\[ x = \sum_i \lambda_i x_i \quad \text{for some } x_i \rightarrow \text{random vector} \]

\[ A^i x_i, A^2 x_i, A^3 x_i, \ldots \]

\[ A^k x_i = \sum_j \lambda_j^k x_j \rightarrow \text{direction tends to } x_1. \]

Tensor decomposition: Tensor Power Method, does not generally work. Only known to work for symmetric orthogonal tensors.

Symmetric if:

\[ T_{ijk} = T_{jik} = T_{ikj} = T_{kij} = T_{kji} = T_{jki} \]

Orthogonal if:

\[ T = \sum_i w_i a_i \otimes a_i \otimes a_i \]

where \( w_i > 0, \ a_i \perp a_j \)

\[ u_0; \text{ iterate } u \rightarrow T(I, u, u) \]

\[ T = \sum_j w_j a_j \otimes a_j \otimes a_j \]

\[ T(I, u, u) = \sum_i w_j (u^T a_j)^2 a_j \]

If \( u = \sum_i a_i a_i \), then this is:

\[ T(I, u, u) = \sum_j w_j a_j^2 a_j \]

(because \( a_i \)'s are orthogonal to each other)
\[ T_{m \times n \times p} \quad T(A', I, I) = \sum_{i=1}^{m} W_i T_{i} \times j \times k A_{i} \times j \]

Where \( W_i = \sum_{i=1}^{m} T_{i} \times j \times k A_{i} \times j \)

Specifically, \( T(n, I, I) = W \) is a \( n \times n \times p \) matrix

with \((i, j, k)\)-th entry \( \sum_{i=1}^{m} W_i T_{i} \times j \times k \)

Tensor Decompositions: Write \( T \) as:

\[ T = \sum_{i}^{q} a_i \otimes b_i \otimes c_i \]

\( a, b, c \) are vectors.

\( R = \) rank of tensor.

- \( R \) can be \( \leq \max(m, n, p) \).
- \( \mathbb{NP} \) hard to determine rank of a tensor.
- Uniqueness:

\[ A = \sum_{j} a_j \otimes b_j \otimes A \times j \times j \times j \times j \]

\( W^T_b \times j = \) some matrix \( A \).

Tensor decompositions are unique under weaker conditions:

\[ T = \sum_{j} a_{j} \otimes b_{j} \otimes c_{j} \]

\( a_{j}, b_{j}, c_{j} = \sum_{j} (a_{j} a_{j}) \otimes (b_{j} b_{j}) \otimes (c_{j} c_{j}) \)

where \( a_{j} b_{j} c_{j} = 1 \).

There are examples where best rank 1 approx.

\( T \) does not appear as a factor in the best rank 2 approx (unlike matrices).
$\alpha_1^2; \alpha_2^2 = \omega_3 (\alpha_1^2)^2$

$\alpha_3^2 = \omega_4 (\alpha_2^2)^2 = \omega_4 (\alpha_1^4)^4$ etc.

---

How do we use tensor decomposition to do MoM for latent variable models?

$\pi = \{ \pi_i \}$

---

1. Example 1: Topic Models, (simplified)

$h_j \text{ (topic)}$

Each $w_j$ is independent of $w_{j'}$, conditioned on $h_j$.

$h \in \{1, \ldots, k\}$, $w_j \in \{ e_1, \ldots, e_d \}$

$\mathbb{P}(h = j) = \pi_j$

$\mathbb{P}(w = e_i | h = j) = \mu_{ij}$

Then:

$\mathbb{E}[ x_1 \otimes x_2 ] = \sum_j \mathbb{P}(x_1 | h = j)^T \mathbb{P}(x_2 | h = j) \mathbb{P}(j)$

$= \sum_j \pi_j \mu_j \mu_j^T$

AND

$\mathbb{E}[ x_1 \otimes x_2 \otimes x_3 ] = \sum_j \mathbb{P}(j) \mathbb{P}(x_1 | h = j) \otimes \mathbb{P}(x_2 | h = j) \otimes \mathbb{P}(x_3 | j)$

$= \sum_j \pi_j \mu_j \otimes \mu_j \otimes \mu_j$

So we can decompose $\mathbb{E}[ x_1 \otimes x_2 \otimes x_3 ]$ sum to get $\mu_j$'s.
Example 2: Multiview Mixture Models.

Mixture of Spherical Gaussians.

\[ h = \text{Mixture component} \]

\[ h \in \{ 1, \ldots, k \}^2 \]

\[ P(h = j) = \pi_j \]

\[ x \mid h = j \sim \mathcal{N}(\mu_j, \sigma^2 I) \]

\[ \mathbb{E}(x \otimes x) = \sum_j \pi_j \mu_j \mu_j^\top + \sigma^2 I. \]

\[ \mathbb{E}(x \otimes x \otimes x) = \sum_j \pi_j \mu_j \otimes \mu_j \otimes \mu_j \]

\[ + \sigma^2 \sum_{i=1}^d (\mathbb{E}(x) \otimes e_i \otimes e_i + e_i \otimes \mathbb{E}(x) \otimes e_i) \]

\[ + e_i \otimes e_i \otimes \mathbb{E}(x) \]

\( \sigma^2 = \min \text{eigenvalue of } \mathbb{E}(x \otimes x) - \mathbb{E}(x) \otimes \mathbb{E}(x) \)

Example 3: Multiview Mixture Models.

\[ h \]

\[ \mu_1, \mu_2, \mu_3 \]

Here, distribution of \( x_6 \) is different from \( x_1 \), but they are conditionally independent given \( h \).
We can write:

$$E(x_t \otimes x_{t+1}) = \sum_{i=1}^{k} \pi_i \mu_{t,i} \otimes \mu_{t+1,i}$$

$$T = E(x_t \otimes x_{t+1} \otimes x_{t+2}) = \sum_{i=1}^{k} \pi_i \mu_{t,i} \otimes \mu_{t+1,i} \otimes \mu_{t+2,i}$$

We can convert $T$ to a symmetric tensor by "symmetrizing"

$$\tilde{x}_1 = E(x_1 \otimes x_2) E(x_1 \otimes x_2)^{-1} x_1$$

$$\tilde{x}_2 = E(x_2 \otimes x_3) E(x_2 \otimes x_3)^{-1} x_2$$

$$\tilde{x}_3 = E(x_3 \otimes x_1) E(x_3 \otimes x_1)^{-1} x_3$$

We can compute $M_2$ : Then,

$$E(\tilde{x}_1 \otimes x_2 \otimes \tilde{x}_3) = \sum_{j} \pi_j \mu_{2,1,j} \otimes \mu_{2,2,j} \otimes \mu_{2,3,j}$$

(Assuming the expectations are of rank $N$).

**Special Case:** Hidden Markov Models

$$\begin{align*}
h_1 & \rightarrow h_2 \rightarrow h_3 \rightarrow \ldots \\
\downarrow & \downarrow \downarrow \\
x_1 & \rightarrow x_2 \rightarrow x_3 \rightarrow \ldots
\end{align*}$$

$$\mu_{2,j} = 0_j \text{ (Observation matrix)}$$
How to convert these into symmetric orthogonal tensor decomposition problems?

Let $W$ be the top $k$ eigenvectors of $\Sigma_{12} = E(X_1 \otimes X_2)$. Then we can decompose and transform back.

$$T(W, W, W) = \sum_{i} \pi_j (W_i \times W_i) \otimes (W_j \times W_j)$$

= symmetric orthogonal tensors.

We can decompose and transform back.

$$p \times (p \otimes p) \otimes (p \otimes p) = p \times$$

$$p \times (p \otimes p) \otimes (p \otimes p) = p \times$$

$\text{Symmetric}$.

$\text{Power}$. The expectations are over some $r$.

Special cases: Hierarchical Markov models.