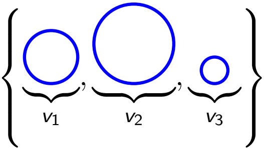
Models in Sparse Coding

Zach Siegel
Advised by Deanna Needell and Guangliang Chen

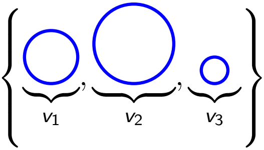
April 4, 2014

$$\left\{ \underbrace{\bigcirc}_{v_1}, \underbrace{\bigcirc}_{v_2}, \underbrace{\circ}_{v_3} \right\}$$



$$\downarrow$$

$$\begin{aligned} v_1 &\to \text{radius} = 1 \\ v_2 &\to \text{radius} = 2 \\ v_3 &\to \text{radius} = 0.5 \end{aligned}$$



$$\downarrow$$

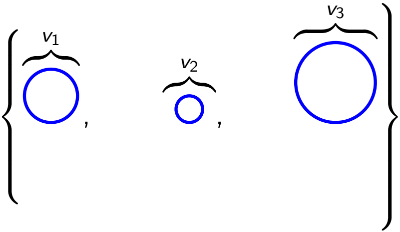
$$\begin{aligned} v_1 &\rightarrow \text{radius} = 1 \\ v_2 &\rightarrow \text{radius} = 2 \\ v_3 &\rightarrow \text{radius} = 0.5 \end{aligned}$$

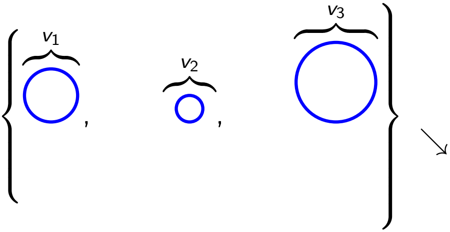
Here's another way:

$$v_1 \rightarrow \begin{pmatrix} \text{radius} = 1 \\ \text{diameter} = 2 \\ \text{circumference} = 2\pi \\ \text{area} = \pi \end{pmatrix} \rightarrow \begin{pmatrix} 1 \\ 2 \\ 2\pi \\ \pi \end{pmatrix},$$

$$v_2 \rightarrow \begin{pmatrix} 2 \\ 4 \\ 4\pi \\ 4\pi \end{pmatrix}$$

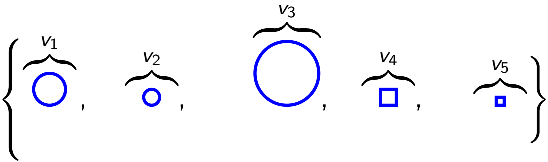
$$v_3 \rightarrow \begin{pmatrix} .5 \\ 1 \\ \pi \\ .25\pi \end{pmatrix}$$


v_1
v_2
v_3

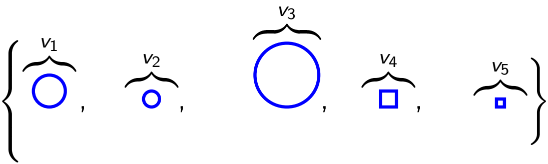


$$\begin{array}{l} \text{radius} = \\ \text{diameter} = \\ \text{circumference} = \\ \text{area} = \end{array} \begin{pmatrix} 1 \\ 2 \\ 2\pi \\ \pi \end{pmatrix} \quad \begin{pmatrix} .5 \\ 1 \\ \pi \\ .25\pi \end{pmatrix} \begin{pmatrix} 2 \\ 4 \\ 4\pi \\ 4\pi \end{pmatrix} \to r \underbrace{\begin{pmatrix} 1 \\ 2 \\ 2\pi \\ 0 \end{pmatrix}}_{b_1} + r^2 \underbrace{\begin{pmatrix} 0 \\ 0 \\ 0 \\ \pi \end{pmatrix}}_{b_2}$$

$$\left\{ \overbrace{\bigcirc}^{v_1}, \quad \overbrace{\circ}^{v_2}, \quad \overbrace{\Large\bigcirc}^{v_3}, \quad \overbrace{\square}^{v_4}, \quad \overbrace{\scriptstyle\square}^{v_5} \right\}$$



$$\begin{array}{l} \text{radius/side} = \\ \text{diameter/diagonal} = \\ \text{circumference/perimeter} = \\ \text{area} = \end{array} \begin{array}{c} \downarrow \\ \begin{pmatrix} 1 \\ 2 \\ 2\pi \\ \pi \end{pmatrix} \end{array} \begin{array}{c} \downarrow \\ \begin{pmatrix} .5 \\ 1 \\ \pi \\ .25\pi \end{pmatrix} \end{array} \begin{array}{c} \downarrow \\ \begin{pmatrix} 2 \\ 4 \\ 4\pi \\ 4\pi \end{pmatrix} \end{array} \begin{array}{c} \downarrow \\ \begin{pmatrix} 1 \\ \sqrt{2} \\ 4 \\ 1 \end{pmatrix} \end{array} \begin{array}{c} \downarrow \\ \begin{pmatrix} .5 \\ \frac{\sqrt{2}}{2} \\ 2 \\ .25 \end{pmatrix} \end{array}$$

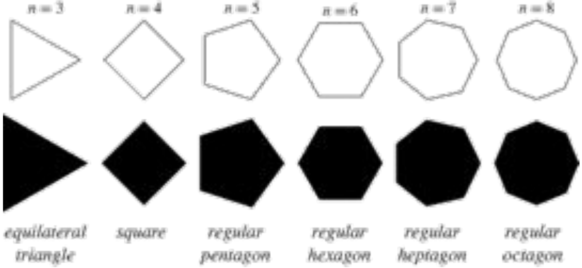


$$\begin{array}{l} \text{radius/side} = \\ \text{diameter/diagonal} = \\ \text{circumference/perimeter} = \\ \text{area} = \end{array} \begin{pmatrix} 1 \\ 2 \\ 2\pi \\ \pi \end{pmatrix} \begin{pmatrix} .5 \\ 1 \\ \pi \\ .25\pi \end{pmatrix} \begin{pmatrix} 2 \\ 4 \\ 4\pi \\ 4\pi \end{pmatrix} \begin{pmatrix} 1 \\ \sqrt{2} \\ 4 \\ 1 \end{pmatrix} \begin{pmatrix} .5 \\ \frac{\sqrt{2}}{2} \\ 2 \\ .25 \end{pmatrix}$$

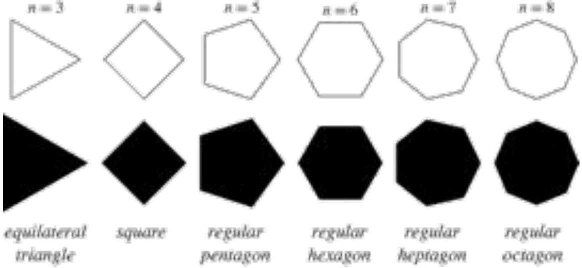
$$\downarrow\downarrow\downarrow \qquad \downarrow\downarrow$$

$$r \underbrace{\begin{pmatrix} 1 \\ 2 \\ 2\pi \\ 0 \end{pmatrix}}_{b_1} + \pi r^2 \underbrace{\begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}}_{b_2}, \quad s \underbrace{\begin{pmatrix} 1 \\ \sqrt{2} \\ 4 \\ 0 \end{pmatrix}}_{b_3} + s^2 \underbrace{\begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}}_{b_2}$$

v_1 v_2 v_3 v_4 v_5 $v_6 \dots$



$$v_1 \quad v_2 \quad v_3 \quad v_4 \quad v_5 \quad v_6 \ldots$$



$$\downarrow \quad \downarrow \quad \downarrow \quad \downarrow \quad \downarrow \quad \downarrow$$

$$\mathcal{B} = \{b_1, b_2, \ldots, b_K\} \subset \mathbb{R}^4$$

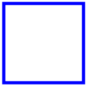
To represent all these different shapes, we need a large collection of bases, and they may not be linearly independent.

- $v_1, \ldots, v_N$ represent shapes
- $\mathcal{B} = \{b_1, \ldots, b_K\}$ each help describe the shapes

- $v_1, \ldots, v_N$ represent shapes
- $\mathcal{B} = \{b_1, \ldots, b_K\}$ each help describe the shapes

For example, a square might be represented

$$v_i = \begin{pmatrix} 1 \\ \sqrt{2} \\ 4 \\ 1 \end{pmatrix} = \big(\text{side length}\big) \underbrace{\begin{pmatrix} 1 \\ \sqrt{2} \\ 4 \\ 0 \end{pmatrix}}_{b_1} + \big(\text{side length}\big)^2 \underbrace{\begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}}_{b_2} \left(= \begin{matrix} \text{side} \\ \text{diagonal} \\ \text{perimeter} \\ \text{area} \end{matrix} \right)$$

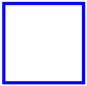


A square

- $v_1, \dots, v_N$ represent shapes
- $\mathcal{B} = \{b_1, \dots, b_K\}$ each help describe the shapes

For example, a square might be represented

$$v_i = \begin{pmatrix} 1 \\ \sqrt{2} \\ 4 \\ 1 \end{pmatrix} = (\text{side length}) \underbrace{\begin{pmatrix} 1 \\ \sqrt{2} \\ 4 \\ 0 \end{pmatrix}}_{b_1} + (\text{side length})^2 \underbrace{\begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}}_{b_2} \left(= \begin{matrix} \text{side} \\ \text{diagonal} \\ \text{perimeter} \\ \text{area} \end{matrix} \right)$$

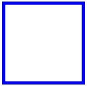


A square

...and we know it's a square because of the choice of bases!!!

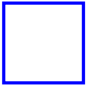
Problem

$$v_i = \begin{pmatrix} 1 \\ \sqrt{2} \\ 4 \\ 1 \end{pmatrix} = \left(\text{side length}\right) \underbrace{\begin{pmatrix} 1 \\ \sqrt{2} \\ 4 \\ 0 \end{pmatrix}}_{b_1} + \left(\text{side length}\right)^2 \underbrace{\begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}}_{b_2} \left(= \begin{array}{c} \text{side} \\ \text{diagonal} \\ \text{perimeter} \\ \text{area} \end{array} \right)$$



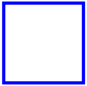
- The support classifies the square
- **We know we can represent every shape as a linear combination of a few of the many bases in $\mathcal{B}$.**
- How do we find the correct support?

We know each shape's description is a linear combination of the bases in $\mathcal{B} = \{b_1, \ldots, b_K\}$:



$$= \begin{pmatrix} 1 \\ \sqrt{2} \\ 4 \\ 1 \end{pmatrix} = v_i = a_1 b_1 + \ldots + a_K b_K = (b_1 | \ldots | b_K) \begin{pmatrix} a_1 \\ \vdots \\ a_K \end{pmatrix}$$

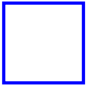
We know each shape's description is a linear combination of the bases in $\mathcal{B} = \{b_1, \ldots, b_K\}$:



$$= \begin{pmatrix} 1 \\ \sqrt{2} \\ 4 \\ 1 \end{pmatrix} = v_i = a_1 b_1 + \ldots + a_K b_K = (b_1 | \ldots | b_K) \begin{pmatrix} a_1 \\ \vdots \\ a_K \end{pmatrix}$$

The "basis" is "overcomplete" - there may be no unique representation

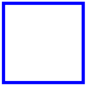
We know each shape's description is a linear combination of the bases in $\mathcal{B} = \{b_1, \ldots, b_K\}$:



$$= \begin{pmatrix} 1 \\ \sqrt{2} \\ 4 \\ 1 \end{pmatrix} = v_i = a_1 b_1 + \ldots + a_K b_K = (b_1 | \ldots | b_K) \begin{pmatrix} a_1 \\ \vdots \\ a_K \end{pmatrix}$$

The "basis" is "overcomplete" - there may be no unique representation

How do we find the representation that tells us v_i is a square?

WANT:  $= a_1 b_1 + a_2 b_2$

SPARSE CODING!

HAVE:

- $B = (b_1 | \ldots | b_K)$ overcomplete basis matrix for $\mathcal{B} = \{b_1, \ldots, b_K\}$
- v_i = description of shape
- $\vec{a}_i = \begin{pmatrix} a_1 \\ \vdots \\ a_K \end{pmatrix}$

WANT: $\square = a_1 b_1 + a_2 b_2$

SOLVE: $\vec{v}_i = B\vec{a}_i$ s.t $||\vec{a}_i||_0 = 2$

SPARSE CODING!

HAVE:

- $B = (b_1 | \ldots | b_K)$ overcomplete basis matrix for $\mathcal{B} = \{b_1, \ldots, b_K\}$
- v_i = description of shape
- $\vec{a}_i = \begin{pmatrix} a_1 \\ \vdots \\ a_K \end{pmatrix}$

WANT: $\boxed{} = a_1 b_1 + a_2 b_2$

SOLVE: $\vec{v}_i = B\vec{a}_i$ s.t $||\vec{a}_i||_0 = 2 \rightarrow \min_{\vec{a}_i} ||v_i - B\vec{a}_i||_2$ s.t. $||\vec{a}_i||_0 \leq 2$

Sparse Coding

SOLVE:

$$\min_{\vec{a}_i} ||v_i - B\vec{a}_i||_2 \text{ s.t. } ||\vec{a}_i||_0 = 2 \text{ or } \leq 2$$

Sparse Coding

SOLVE:

$$\min_{\vec{a}_i} ||v_i - B\vec{a}_i||_2 \text{ s.t. } ||\vec{a}_i||_0 = 2 \text{ or } \leq 2$$

or

$$\min_{\vec{a}_i} ||v_i - B\vec{a}_i||_2^2 + \lambda ||\vec{a}_i||_0$$

Sparse Coding

SOLVE:

$$\min_{\vec{a}_i} ||v_i - B\vec{a}_i||_2 \text{ s.t. } ||\vec{a}_i||_0 = 2 \text{ or } \leq 2$$

or

$$\min_{\vec{a}_i} ||v_i - B\vec{a}_i||_2^2 + \lambda||\vec{a}_i||_0$$

This $||\cdot||_0$ is non-linear!

Sparse Coding

$$\min_{\vec{a}_i} ||v_i - B\vec{a}_i||_2^2 + \lambda||\vec{a}_i||_0$$

- Finding $v_i \approx B\vec{a}_i$ is easy, $\mathcal{B}$ is overcomplete (underdetermined)
- Finding a *sparse* solution is difficult, non-linear

But the sparse solution allows us to classify data!!!

Sparse Coding

Definition

The *support* of a vector is the indices of its nonzero entries

Definition

The *support vector* of $\vec{v}$ is $\vec{s}$ s.t. $\vec{s}_i = \begin{cases} 0 & : \quad v_i = 0 \\ 1 & : \quad \text{otherwise} \end{cases}$

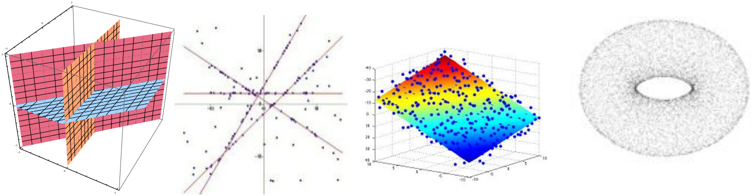
MATLAB: $s = (abs(v) > 0)$

So sparse representations are hard to find, but allow data classification - "SQUARE" ness was all in the support

Geometric Interpretation

Sparse representations constrain representation to union of planes spanned by a few bases in $\mathcal{B}$.

Where does the data live?



On the other hand, the SPAN of $\mathcal{B}$ is the entire ambient space

Sparse Data

We only want to think about data as sparse if it is ACTUALLY on the union of some planes in its ambient space.

What other data is sparse?

Sparse Data

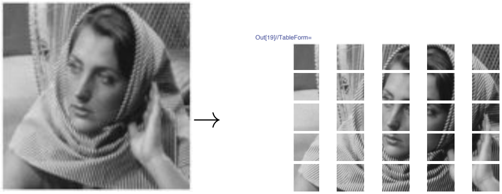
We only want to think about data as sparse if it is ACTUALLY on the union of some planes in its ambient space.

What other data is sparse?

A TON OF STUFF!

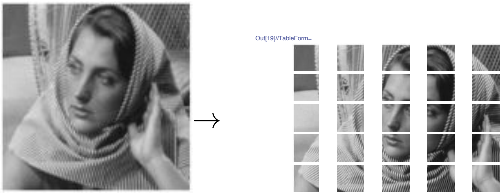
Why are Images Sparse?

Divide an image into patches:

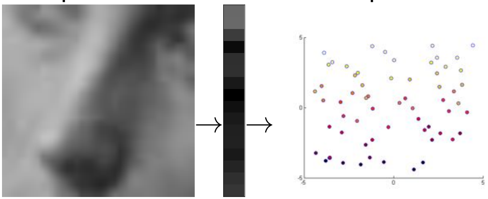


Why are Images Sparse?

Divide an image into patches:

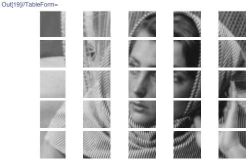


Turn patches into vectors in a point cloud:



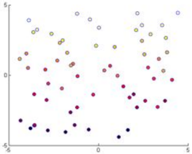
Why are Images Sparse?

Divide an image into patches:


→


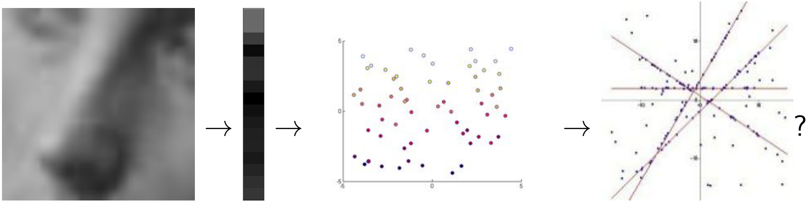
Turn patches into vectors in a point cloud:


→

→


Why would these vectors lie on a union of planes??? Couldn't they lie "anywhere"?

Why are Images Sparse?



Small natural image patches DO lie on a union of subspaces!
Structure is low-dimensional.

(Argument using hands [about corners, etc])
(Argument using pictures [examples])

So, image patches are inherently 'sparse' (live on a union of low-dimensional subspaces).

In the same way as the shapes, **image patches can be classified by feature!**
(Not true of non-sparse representations)

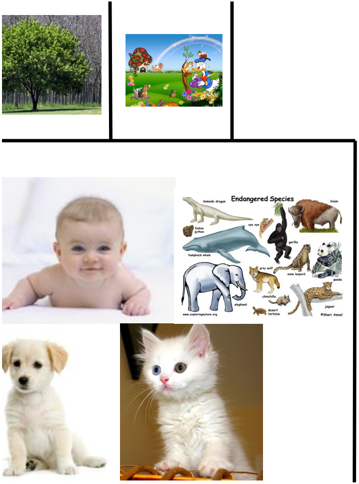
So, image patches are inherently ‘sparse’ (live on a union of low-dimensional subspaces).

In the same way as the shapes, **image patches can be classified by feature!**
(Not true of non-sparse representations)

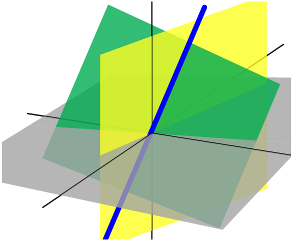
How can we partition the space in which image patches lie to classify the patches?

Structured Sparsity

Block Sparsity



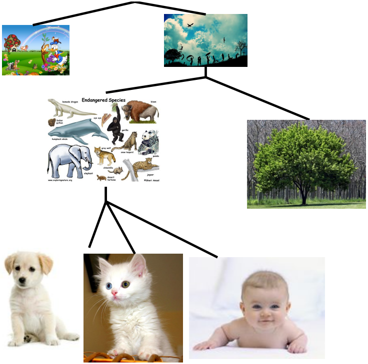
- Animal image subspace
- Foliage image subspace
- Cartoon image subspace



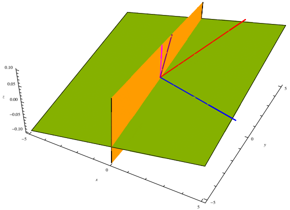
Like the shapes example!

Structured Sparsity

Hierarchical Sparsity



- Cartoon subspace
- baby, cat, dog subspaces $\subseteq$ animal subspace
- animal, foliage subspaces $\subseteq$ natural image subspace



How can we induce this kind of structure in our sparsity?

Recall the objective function:

$$\min_{\vec{a}} ||v - B\vec{a}||_2^2 + \lambda||\vec{a}||_0$$

How can we induce this kind of structure in our sparsity?

Recall the objective function:

$$\min_{\vec{a}} || \underbrace{v}_{\substack{\text{data}\\ \text{point}}} - \underbrace{B}_{\substack{\text{overcomplete}\\ \text{basis}\\ \text{matrix}}} \times \underbrace{\vec{a}}_{\text{representation}} ||_2^2 + \underbrace{\lambda||\vec{a}||_0}_{\substack{\text{sparsity}\\ \text{penalty}}}$$

How can we induce this kind of structure in our sparsity?

Recall the objective function:

$$\min_{\vec{a}} || \underbrace{v}_{\substack{\text{data}\\\text{point}}} - \underbrace{B}_{\substack{\text{overcomplete}\\\text{basis}\\\text{matrix}}} \times \underbrace{\vec{a}}_{\text{representation}} ||_2^2 + \underbrace{\lambda||\vec{a}||_0}_{\substack{\text{sparsity}\\\text{penalty}}}$$

Question: How can we favor block sparsity or hierarchical sparsity?

Rephrase the problem:

$$\min_{\vec{a}} ||\vec{v} - B\vec{a}||_2^2 + \lambda||\vec{a}||_0$$
$$\downarrow$$

$P(\vec{a}$ is the best solution$|v)$ **decreases** w/r/t $||v - B\vec{a}||_2^2$ and $||\vec{a}||_0$

Rephrase the problem:

$$\min_{\vec{a}} ||\vec{v} - B\vec{a}||_2^2 + \lambda||\vec{a}||_0$$
$$\downarrow$$

$$\max_{\vec{a}} \overbrace{\underbrace{P(\vec{a} \text{ is the best solution}|v)}_{\text{posterior!}}}^{\text{New objective f'n}} \textbf{ decreases } \text{w/r/t } ||v - B\vec{a}||_2 \text{ and } ||\vec{a}||_0$$

(Note that $\vec{a}$ is a parameter of v)

Rephrase the problem:

$$\min_{\vec{a}} ||\vec{v} - B\vec{a}||_2^2 + \lambda||\vec{a}||_0$$
$$\downarrow$$

$$\max_{\vec{a}} \overbrace{\underbrace{P(\vec{a} \text{ is the best solution}|v)}_{\text{posterior!}}}^{\text{New objective f'n}} \textbf{ decreases } \text{w/r/t } ||v - B\vec{a}||_2 \text{ and } ||\vec{a}||_0$$

(Note that $\vec{a}$ is a parameter of v)

Can we define a prior $P(\vec{a})$? And add structure to favor block/hierarchical sparsity?

Rephrase the problem:

$$\min_{\vec{a}} ||\vec{v} - B\vec{a}||_2^2 + \lambda||\vec{a}||_0$$
$$\downarrow$$

$$\max_{\vec{a}} \overbrace{\underbrace{P(\vec{a} \text{ is the best solution}|v)}_{\text{posterior!}}}^{\text{New objective f'n}} \textbf{ decreases } \text{w/r/t } ||v - B\vec{a}||_2 \text{ and } ||\vec{a}||_0$$

(Note that $\vec{a}$ is a parameter of v)

Can we define a prior $P(\vec{a})$? And add structure to favor block/hierarchical sparsity?
Can we then define the likelihood $P(\vec{v}|\vec{a})$?

Rephrase the problem:

$$\min_{\vec{a}} ||\vec{v} - B\vec{a}||_2^2 + \lambda||\vec{a}||_0$$

$$\downarrow$$

$$\max_{\vec{a}} \overbrace{\underbrace{P(\vec{a} \text{ is the best solution}|v)}_{\text{posterior!}}}^{\text{New objective f'n}} \textbf{ decreases } \text{w/r/t } ||v - B\vec{a}||_2 \text{ and } ||\vec{a}||_0$$

(Note that $\vec{a}$ is a parameter of v)

Can we define a prior $P(\vec{a})$? And add structure to favor block/hierarchical sparsity?
Can we then define the likelihood $P(\vec{v}|\vec{a})$?
Can we maximize the posterior $P(\vec{a}|\vec{v})$ (like we minimized the simpler objective function)?

Rephrase the problem:

$$\min_{\vec{a}} ||\vec{v} - B\vec{a}||_2^2 + \lambda||\vec{a}||_0$$

$$\downarrow$$

$$\max_{\vec{a}} \overbrace{\underbrace{P(\vec{a} \text{ is the best solution}|v)}_{\text{posterior!}}}^{\text{New objective f'n}} \textbf{ decreases } \text{w/r/t } ||v - B\vec{a}||_2 \text{ and } ||\vec{a}||_0$$

(Note that $\vec{a}$ is a parameter of v)

Can we define a prior $P(\vec{a})$? And add structure to favor block/hierarchical sparsity?
Can we then define the likelihood $P(\vec{v}|\vec{a})$?
Can we maximize the posterior $P(\vec{a}|\vec{v})$ (like we minimized the simpler objective function)?
YES

Prior for Support

$\vec{s}$ is support vector of $\vec{a}$: $\vec{s}_i = \begin{cases} 0 & : \quad a_i = 0 \\ 1 & : \quad \text{otherwise} \end{cases}$

$\vec{v} \approx B\vec{a}$

$P(\vec{a}|\vec{v}) \propto P(\vec{v}|\vec{a}) \underbrace{P(\vec{a}|\vec{s})P(\vec{s})}_{P(a)}$

Prior for Support

$\vec{s}$ is support vector of $\vec{a}$: $\vec{s}_i = \begin{cases} 0 & : \quad a_i = 0 \\ 1 & : \quad \text{otherwise} \end{cases}$

$\vec{v} \approx B\vec{a}$

$P(\vec{a}|\vec{v}) \propto P(\vec{v}|\vec{a}) \underbrace{P(\vec{a}|\vec{s})P(\vec{s})}_{P(a)}$

- $P(\vec{s}) = P(\text{support}(\vec{a}))$
  Decreases with $||\vec{s}||_0 = ||\vec{a}||_0$ and with lack-of-structure
- $P(\vec{a}|\vec{s}) = P(\vec{a}|\text{support}(\vec{a}))$
  $(\vec{a}_i|\vec{s}_i = 1) \sim N(0, \sigma)$ (remember - this is not so important)

Prior for Support

$\vec{s}$ is support vector of $\vec{a}$: $\vec{s}_i = \begin{cases} 0 & : \quad a_i = 0 \\ 1 & : \quad \text{otherwise} \end{cases}$

$\vec{v} \approx B\vec{a}$

$P(\vec{a}|\vec{v}) \propto P(\vec{v}|\vec{a}) \underbrace{P(\vec{a}|\vec{s})P(\vec{s})}_{P(a)}$

- $P(\vec{s}) = P(\text{support}(\vec{a}))$
  Decreases with $||\vec{s}||_0 = ||\vec{a}||_0$ and with lack-of-structure
- $P(\vec{a}|\vec{s}) = P(\vec{a}|\text{support}(\vec{a}))$
  $(\vec{a}_i|\vec{s}_i = 1) \sim N(0, \sigma)$ (remember - this is not so important)
- $P(\vec{v}|\vec{a})$ decreases with $||\vec{v} - B\vec{a}||_2^2$

Penalize $||\vec{s}||_0 = ||\vec{a}||_0$ and Reward Structure

$$\vec{s} \text{ binary} \Rightarrow ||\vec{s}||_0 = \underbrace{(1, \ldots, 1)}_{K} \cdot \begin{pmatrix} s_i \\ \vdots \\ s_K \end{pmatrix} \rightarrow \underbrace{(\text{bias}_1, \ldots, \text{bias}_K)}_{\vec{bias}} \cdot \begin{pmatrix} s_i \\ \vdots \\ s_K \end{pmatrix}$$

Multiplication by penalty vector $\vec{bias}$ in place of $|| \cdot ||_0$.

Penalize $||\vec{s}||_0 = ||\vec{a}||_0$ and Reward Structure

$$\vec{s} \text{ binary} \Rightarrow ||\vec{s}||_0 = \underbrace{(1,\ldots,1)}_{K} \cdot \begin{pmatrix} s_i \\ \vdots \\ s_K \end{pmatrix} \rightarrow \underbrace{(\mathrm{bias}_1,\ldots,\mathrm{bias}_K)}_{\vec{bias}} \cdot \begin{pmatrix} s_i \\ \vdots \\ s_K \end{pmatrix}$$

Multiplication by penalty vector $\vec{bias}$ in place of $||\cdot||_0$.

To penalize 'second-order' structure (like blocks/hierarchy),

$$\vec{s}^T W \vec{s} = \sum_i \sum_j w_{ij} s_i s_j$$

SO the support's structure encoded in $\vec{bias}$ and interactions matrix W!!!

Penalize $||\vec{s}||_0 = ||\vec{a}||_0$ and Reward Structure

$$\vec{s} \text{ binary} \Rightarrow ||\vec{s}||_0 = \underbrace{(1,\ldots,1)}_{K} \cdot \begin{pmatrix} s_i \\ \vdots \\ s_K \end{pmatrix} \rightarrow \underbrace{(\text{bias}_1,\ldots,\text{bias}_K)}_{\vec{bias}} \cdot \begin{pmatrix} s_i \\ \vdots \\ s_K \end{pmatrix}$$

Multiplication by penalty vector $\vec{bias}$ in place of $||\cdot||_0$.

To penalize 'second-order' structure (like blocks/hierarchy),

$$\vec{s}^T W \vec{s} = \sum_i \sum_j w_{ij} s_i s_j$$

SO the support's structure encoded in $\vec{bias}$ and interactions matrix W!!!
!!!!!

$$P(s) \propto e^{\vec{bias}\cdot\vec{s} + \vec{s}^T W \vec{s}}$$

BOLTZMANN MACHINE DISTRIBUTION

$$P(s) \propto e^{\vec{bias}\cdot\vec{s}+\vec{s}^T W \vec{s}}$$

This is the Boltzmann Matchine distribution, and now you get what it's all about!

Posterior for $\vec{a}$

$$P(\vec{a}|\vec{v}) \propto P(\vec{s}) \times P(\vec{a}|\vec{s}) \times P(\vec{v}|\vec{a})$$

Posterior for $\vec{a}$

$$P(\vec{a}|\vec{v}) \propto \underbrace{P(\vec{s})}_{\propto e^{\vec{bias}\cdot\vec{s}+\vec{s}^T W\vec{s}}} \times P(\vec{a}|\vec{s}) \times P(\vec{v}|\vec{a})$$

Posterior for $\vec{a}$

$$P(\vec{a}|\vec{v}) \propto \underbrace{P(\vec{s})}_{\propto e^{\vec{bias}\cdot\vec{s}+\vec{s}^T W\vec{s}}} \times \underbrace{P(\vec{a}|\vec{s})}_{\substack{\sim N(0,\sigma) \\ \text{(not important)}}} \times P(\vec{v}|\vec{a})$$

Posterior for $\vec{a}$

$$P(\vec{a}|\vec{v}) \propto \overbrace{\underbrace{P(\vec{s})}_{\propto e^{\vec{bias}\cdot\vec{s}+\vec{s}^T W \vec{s}}} \times \underbrace{P(\vec{a}|\vec{s})}_{\substack{\sim N(0,\sigma) \\ \text{(not important)}}}}^{P(\vec{a})} \times \underbrace{P(\vec{v}|\vec{a})}_{\substack{\text{Decreases} \\ \text{with} \\ ||\vec{v} - B\vec{a}||_2}}$$

Posterior for $\vec{a}$

$$\underbrace{P(\vec{a}|\vec{v})}_{\substack{\text{Objective} \\ \text{function}}} \propto \overbrace{\underbrace{P(\vec{s})}_{\propto e^{\vec{bias}\cdot\vec{s}+\vec{s}^T W\vec{s}}} \times \underbrace{P(\vec{a}|\vec{s})}_{\substack{\sim N(0,\sigma) \\ \text{(not important)}}}}^{P(\vec{a})} \times \underbrace{P(\vec{v}|\vec{a})}_{\substack{\text{Decreases} \\ \text{with} \\ ||\vec{v}-B\vec{a}||_2}}$$

Posterior for $\vec{a}$

$$\underbrace{P(\vec{a}|\vec{v})}_{\substack{\text{Objective}\\ \text{function}}} \propto \overbrace{\underbrace{P(\vec{s})}_{\propto e^{\vec{bias}\cdot\vec{s}+\vec{s}^T W\vec{s}}} \times \underbrace{P(\vec{a}|\vec{s})}_{\substack{\sim N(0,\sigma)\\ \text{(not important)}}}}^{P(\vec{a})} \times \underbrace{P(\vec{v}|\vec{a})}_{\substack{\text{Decreases}\\ \text{with}\\ ||\vec{v}-B\vec{a}||_2}}$$

Moral: We now have a wonderful new objective function whose optimization entails structured sparsity thanks to the interactions matrix W.