# DSC 190/291 · Assignment 5 Solutions

UCSD · Spring 2026

---

## Part A: Sparse Linear Predictors: Statistical Sparsity versus Agnostic Hardness (45 points)

This problem continues the sparse-linear-model theme from Homework 4. There, you studied sparse halfspaces through VC dimension, structural risk minimization, validation, and brute-force search over supports. Here you will see what changes in the agnostic proper setting: the statistical story survives, but the empirical 0-1 optimization problem can become computationally hard when the sparsity level is part of the input.

For $d \in \mathbb{N}$ and $k \leq d$, define the class of $k$-sparse homogeneous linear classifiers

$$\mathcal{H}_{d,k} = \{h_w : x \to \text{sign}(\langle w, x \rangle) : w \in \mathbb{R}^d, \|w\|_0 \leq k\},$$

where throughout this problem

$$\text{sign}(t) = +1$$

if $t \geq 0$, and $\text{sign}(t) = -1$ otherwise.

For a labeled sample $S = ((x_1, y_1), ..., (x_n, y_n))$, define

$$\textsf{AGREEMENT}_{\mathcal{H}_{d,k}}(S, K) = 1$$

iff there exists $h_w \in \mathcal{H}_{d,k}$ that correctly labels at least $K$ examples in $S$.

1. **(7 points) What Homework 4 already gives you.**

   Recall from Homework 4 that

   $$\text{VCdim}(\mathcal{H}_{d,k}) \leq O\left(k \log\left(e\frac{d}{k}\right)\right)$$

   for $1 \leq k \leq \frac{d}{2}$. For larger $k$, the usual dense-halfspace bound gives $\text{VCdim}(\mathcal{H}_{d,k}) \leq O(d)$. Do not reprove this bound. Explain what it implies for agnostic sample complexity if exact empirical risk minimization over $\mathcal{H}_{d,k}$ were computationally available.

   Also recall the brute-force realizable algorithm from Homework 4: enumerate supports $T \subseteq [d]$ with $|T| \leq k$, and for each support solve a linear feasibility problem. State its runtime up to polynomial factors in the feasibility solver. In which regimes of $k$ is this polynomial in $d$? Explain why this is a statistical/computational distinction rather than a contradiction.

   **Solution.**

   If exact ERM over $\mathcal{H}_{d,k}$ were available, the agnostic VC theorem would give

   $$L_{\mathcal{D}}(\hat{h}) \leq \inf_{h \in \mathcal{H}_{d,k}} L_{\mathcal{D}}(h) + \varepsilon$$

with probability at least $1-\delta$ from

$$n = O\left(\frac{\min\{k\log(e\frac{d}{k}), d\} + \log(\frac{1}{\delta})}{\varepsilon^2}\right)$$

samples, up to universal constants. Thus the statistical sample complexity reflects the sparse VC dimension when $k \le \frac{d}{2}$, and never exceeds the dense halfspace rate of order $d$.

The brute-force realizable algorithm enumerates all supports of size at most $k$ and, for each support, solves a homogeneous halfspace feasibility problem in that support. The number of supports is

$$\sum_{s=0}^{k}\binom{d}{s} \le \left(e\frac{d}{k}\right)^k$$

for $1 \le k \le d$. Therefore the runtime is

$$\left(e\frac{d}{k}\right)^k \mathrm{poly}(n,k)$$

up to the cost of the linear feasibility solver.

This is polynomial in $d$ when $k$ is fixed. When $k$ grows with $d$, the enumeration is generally not polynomial in $d$; for example $k = O(\log d)$ gives quasi-polynomial-type behavior. There is no contradiction: VC dimension controls how many samples are statistically sufficient, while the brute-force runtime reflects the computational cost of searching over supports.

2. **(24 points) Hardness of agreement via set cover.**

   Reduce SETCOVER to $\mathsf{AGREEMENT}_{\mathcal{H}_{d,k}}$. A SETCOVER instance consists of a universe $U = \{u_1, ..., u_r\}$, sets $A_1, ..., A_q \subseteq U$, and an integer $k \le q$. The question is whether there exists $R \subseteq [q]$ with $|R| \le k$ and $\bigcup_{j\in R} A_j = U$.

   Construct a labeled sample and threshold $K$ such that the set-cover instance is a yes-instance iff there is a $k$-sparse homogeneous linear classifier that correctly labels at least $K$ examples. The sample may be treated as a multiset, so repeated examples count with multiplicity.

   Hint: use one coordinate per set. Think of negative coordinates of $w$ as the selected sets. Positive examples should charge the number of selected sets, while repeated negative examples should reward covered universe elements. You may use $M = q + 1$ repeated examples if helpful.

   Prove both directions of the reduction.

   Then invoke the Week 5 learner-to-agreement theorem. Explain in two or three sentences how it applies to the family $\mathcal{H}_{d,k}$ when both $d$ and $k$ are part of the input. Conclude that, if $\mathsf{NP} \neq \mathsf{RP}$, there is no efficient proper agnostic PAC learner for sparse linear classifiers when $k$ is part of the input.

   **Solution.**

Set $d = q$, one coordinate for each set $A_j$. Let $e_j \in \mathbb{R}^q$ be the $j$th standard basis vector. For each universe element $u_l$, define

$$a_l \in \{0,1\}^q, \quad (a_l)_j = \mathbf{1}\big[u_l \in A_j\big].$$

Let $M = q + 1$.

The labeled multiset $S$ contains:

- for each set index $j \in [q]$, one positive example $\big(e_j, +1\big)$;
- for each universe element $u_l$, $M$ copies of the negative example $(a_l, -1)$.

Set

$$K = q - k + Mr.$$

First suppose the set-cover instance is a yes-instance. Let $R \subseteq [q]$ be a cover with $|R| \le k$. Define

$$w_j = -1$$

if $j \in R$ and $w_j = 0$ otherwise. Then $w$ is $k$-sparse. On the positive example $e_j$, the classifier is wrong exactly when $j \in R$, because

$$\langle w, e_j \rangle = w_j$$

and $\operatorname{sign}(0) = +1$. Thus it correctly labels $q - |R| \ge q - k$ positive examples. For every universe element $u_l$, some selected set $A_j$ contains it, so

$$\langle w, a_l \rangle = -\big|\big\{j \in R : u_l \in A_j\big\}\big| < 0.$$

Hence all $Mr$ repeated negative examples are correctly labeled. The total number of correct labels is at least $q - k + Mr = K$.

Conversely, suppose some $k$-sparse $w$ correctly labels at least $K$ examples. Let

$$R = \big\{j : w_j < 0\big\}.$$

Since $w$ is $k$-sparse, $|R| \le k$. A positive example $\big(e_j, +1\big)$ is mislabeled exactly when $w_j < 0$, so the number of correctly labeled positive examples is $q - |R|$.

Let $C$ be the number of universe elements whose negative example $a_l$ is correctly labeled. If $a_l$ is correctly labeled, then

$$\langle w, a_l \rangle < 0.$$

Since the coordinates of $a_l$ are nonnegative, this implies that some coordinate $j$ with $u_l \in A_j$ has $w_j < 0$. Therefore the correctly labeled universe elements are covered by $R$.

The total number of correct labels is exactly

$$q - |R| + MC.$$

Since this is at least $K = q - k + Mr$,

$$q - |R| + MC \ge q - k + Mr,$$

so

$$M(r - C) \le k - |R| \le k \le q.$$

But $M = q + 1$. Therefore $r - C$ cannot be positive, and hence $C = r$. All universe elements are covered by $R$, so the set-cover instance is a yes-instance.

This is a polynomial-time many-one reduction from SETCOVER to $\mathsf{AGREEMENT}_{\mathcal{H}_{d,k}}$ with $d = q$, while $k$ remains the set-cover budget. Thus both $d$ and $k$ are part of the generated agreement instance. Since SETCOVER is NP-hard, agreement for sparse homogeneous halfspaces is NP-hard when both $d$ and $k$ are part of the input. By the Week 5 learning-to-agreement theorem, an efficient proper agnostic PAC learner for this family would imply that the agreement problem is in RP. Therefore, assuming $\mathsf{NP} \neq \mathsf{RP}$, no efficient proper agnostic PAC learner exists for sparse linear classifiers when the sparsity level is part of the input.

3. **(14 points) Experiment and interpretation.**

   Implement a small experiment comparing proper sparse 0-1 search with a convex surrogate method. Use synthetic data of your choice.

   Your experiment should include:

   - a proper sparse 0-1 search baseline for small $d$ and $k$. For example, enumerate supports $T$ with $|T| \le k$, then either enumerate a finite grid of weights on each support or use any exact small-sample method you can justify. State clearly whether your baseline is exact or approximate;
   - a convex surrogate method, for example logistic regression or hinge-loss minimization with an $\ell_1$ constraint or penalty;
   - a comparison of runtime scaling as $d$ or $k$ increases;
   - a comparison of empirical 0-1 agreement and the surrogate objective or validation error.

   In your write-up, explain what the experiment illustrates about the difference between empirical 0-1 agreement over $k$-sparse classifiers and convex surrogate optimization over an $\ell_1$-controlled linear predictor. Your explanation should also say whether the surrogate method returns a proper $k$-sparse classifier.

   **Solution.**

   One possible experiment is the following.

   Generate data by choosing a sparse vector $w^*$ with support size $k^*$, drawing $x_i \sim \mathcal{N}(0, I_d)$, setting $y_i = \operatorname{sign}(\langle w^*, x_i \rangle)$, and then flipping a small fraction of labels. Split the data into training and validation sets.

   For the proper sparse 0-1 baseline, enumerate all supports $T \subseteq [d]$ with $|T| \le k$. On each support, enumerate a finite grid of weights, for example values in $\{-2, -1, 1, 2\}$ on the selected coordinates, and choose the vector with largest training agreement. This baseline is approximate for the continuous sparse halfspace class, because it searches only a finite grid of weights. Its number of candidates is

   $$\sum_{s=0}^{k} \binom{d}{s} 4^s,$$

which grows quickly with both $d$ and $k$.

For the convex surrogate method, fit logistic regression with an $\ell_1$ penalty, or minimize hinge loss with an $\ell_1$ constraint. This solves a convex surrogate problem over an $\ell_1$ -controlled linear predictor. The method can be run for the same values of $d$ and evaluated by training agreement, validation error, and surrogate objective value.

A typical comparison is:

- for fixed small $k$, the sparse grid baseline becomes slower as $d$ grows roughly like $d^k$;
- increasing $k$ is much more expensive because it increases the support search space;
- the convex surrogate scales more smoothly because it does not enumerate supports;
- the sparse baseline directly optimizes empirical 0-1 agreement over the searched grid;
- the surrogate method optimizes a different objective, so the lowest surrogate value need not correspond to the best sparse 0-1 agreement.

The surrogate output is not, in general, a proper $k$-sparse classifier. An $\ell_1$ penalty encourages sparsity, but it does not enforce exactly $\|w\|_0 \le k$. Even when many coordinates are small, the output belongs to the convex $\ell_1$-controlled class, not necessarily to $\mathcal{H}_{d,k}$.

---

## Part B: Convex Fixed-Feature Learning and Its Limits (40 points)

This problem studies the fixed-feature convex-learning template and its limits. First, you will see how hinge-loss minimization with an $\ell_1$ constraint becomes a linear program. Second, you will show that optimizing a surrogate need not optimize the best 0-1 classifier. Third, you will prove the fixed-feature parity barrier stated in lecture. Finally, you will see why learning the feature map, as neural networks do, destroys convexity.

In Part B, the 0-1 margin loss counts zero margin as an error. This is separate from Part A's tie-breaking convention $\mathrm{sign}(0) = +1$ for sparse classifiers.

For binary labels $y \in \{-1, +1\}$, define the 0-1 loss of a real-valued score $f(x)$ as

$$\ell_{0\text{-}1}(f(x), y) = \mathbf{1}[yf(x) \le 0],$$

and define the hinge loss as

$$\ell_{\text{hinge}}(f(x), y) = (1 - yf(x))_+.$$

1. **(10 points) Hinge loss with an $\ell_1$ constraint as a linear program.**

   A **linear program** is an optimization problem with a linear objective and linear equality or inequality constraints. For example, the variables might be $z_1, ..., z_N$, the objective might be $\min \sum_j c_j z_j$, and the constraints might have the form $\sum_j a_{ij} z_j \le b_i$. The word **linear** means that variables are only multiplied by constants and added together; expressions such as $z_i z_j$, $z_i^2$, $|z_i|$, and $(z_i)_+$ are not themselves linear. A common trick is to introduce extra variables and linear constraints whose optimum reproduces a non-linear-looking expression.

   Let $f_w(x) = \langle w, x \rangle$ with $w \in \mathbb{R}^d$. Fix a radius $B > 0$ and consider

$$\min_{w:\ \|w\|_1 \le B} \frac{1}{n} \sum_{i=1}^{n} (1 - y_i \langle w, x_i \rangle)_+.$$

Introduce variables $\xi_i \ge 0$ for the hinge losses and nonnegative variables $w_j^+, w_j^-$ with $w_j = w_j^+ - w_j^-$. Write a linear program whose optimal value equals the optimization problem above. Explain the role of each family of constraints.

Then explain why this linear program is not solving empirical 0-1 agreement over $k$-sparse classifiers.

**Solution.**

Use variables $w_j^+, w_j^- \ge 0$ for $j = 1, ..., d$ and $\xi_i \ge 0$ for $i = 1, ..., m$. The linear program is

$$\min_{w^+, w^-, \xi} \frac{1}{n} \sum_{i=1}^{n} \xi_i$$

subject to

$$\xi_i \ge 1 - y_i \sum_{j=1}^{d} \left(w_j^+ - w_j^-\right) x_{ij} \quad \text{for all } i,$$

$$\xi_i \ge 0 \quad \text{for all } i,$$

$$\sum_{j=1}^{d} \left(w_j^+ + w_j^-\right) \le B,$$

$$w_j^+ \ge 0, \quad w_j^- \ge 0 \quad \text{for all } j.$$

The variables $w_j^+$ and $w_j^-$ represent the positive and negative parts of $w_j$, so $w_j = w_j^+ - w_j^-$. The constraint $\sum_j \left(w_j^+ + w_j^-\right) \le B$ guarantees $\|w\|_1 \le B$ at every feasible point, and at optimum we may take $w_j^+ + w_j^- = |w_j|$. For fixed $w$, the constraints on $\xi_i$ force

$$\xi_i \ge (1 - y_i \langle w, x_i \rangle)_+,$$

and because the objective minimizes the sum of the $\xi_i$, equality holds at optimum.

This is not empirical 0-1 agreement over $k$-sparse classifiers. It minimizes hinge loss rather than the number of mistakes, and the feasible set is an $\ell_1$ ball rather than the nonconvex sparsity constraint $\|w\|_0 \le k$. The resulting classifier need not be a proper member of $\mathcal{H}_{d,k}$.

2. **(10 points) A concrete counterexample: the surrogate comparator can be bad.**

   Fix $p \in \left(0, \frac{1}{2}\right)$ and $M > \frac{1-p}{p}$. Consider homogeneous one-dimensional linear predictors $f_w(x) = wx$ and the distribution $\mathcal{D}_{p,M}$ supported on two labeled examples:

   $$\mathbb{P}[(x, y) = (1, +1)] = 1 - p, \quad \mathbb{P}[(x, y) = (-M, +1)] = p.$$

   Compute the population hinge risk as a piecewise-linear function of $w$, and determine all of its minimizers. Then prove the following statements exactly:

   - $\inf_w L_{\mathcal{D}}^{0\text{-}1}(f_w) = p$;

- every population hinge-risk minimizer has 0-1 risk equal to $1-p$.

Conclude that for any $\varepsilon > 0$ and any $\alpha < 1$, there are $p, M$ such that

$$\inf_w L_{\mathcal{D}}^{0\text{-}1}(f_w) \le \varepsilon$$

but every hinge-risk minimizer has 0-1 risk greater than $\alpha$.

This example is intentionally low-dimensional. The difficulty is not algebraic; it is recognizing exactly what the surrogate objective is optimizing.

**Solution.**

The population hinge risk is

$$L_{\mathcal{D}}^{\text{hinge}}(f_w) = (1-p)(1-w)_+ + p(1+Mw)_+.$$

The breakpoints are $w = -\frac{1}{M}$ and $w = 1$. Thus

$$L_{\mathcal{D}}^{\text{hinge}}(f_w) = \begin{cases} (1-p)(1-w) \\ w \le -\frac{1}{M} \\ (1-p)(1-w) + p(1+Mw) \\ -\frac{1}{M} \le w \le 1 \\ p(1+Mw) \\ w \ge 1. \end{cases}$$

On the first interval, the slope is $-(1-p) < 0$, so the risk decreases up to $-\frac{1}{M}$. On the middle interval, the slope is $pM - (1-p) > 0$ by the assumption on $M$, so the risk increases after $-\frac{1}{M}$. On the third interval, the slope is $pM > 0$. Therefore the unique population hinge-risk minimizer is

$$w = -\frac{1}{M}.$$

Now compute the 0-1 risk, remembering that zero margin is an error in Part B. If $w > 0$, then $f_w(1) > 0$ and $f_w(-M) < 0$, so the example $(1, +1)$ is correct and $(-M, +1)$ is wrong. The 0-1 risk is $p$. If $w < 0$, then $(1, +1)$ is wrong and $(-M, +1)$ is correct, so the 0-1 risk is $1-p$. If $w = 0$, both margins are zero and both examples are errors, so the risk is 1.

Hence

$$\inf_w L_{\mathcal{D}}^{0\text{-}1}(f_w) = p,$$

achieved by any $w > 0$. The unique hinge-risk minimizer $w = -\frac{1}{M}$ has 0-1 risk $1-p$.

Given any $\varepsilon > 0$ and $\alpha < 1$, choose $0 < p < \min\{\frac{1}{2}, \varepsilon, 1-\alpha\}$ and then choose $M > \frac{1-p}{p}$. Then $\inf_w L_{\mathcal{D}}^{0\text{-}1}(f_w) = p \le \varepsilon$ while every hinge-risk minimizer has 0-1 risk $1-p > \alpha$.

3. **(12 points) The fixed-feature parity barrier: proof and tightness.**

Work on $\mathcal{X} = \{-1, +1\}^d$. For each subset $I \subseteq [d]$, define the parity function

$$\chi_I(x) = \prod_{i \in I} x_i.$$

In lecture, we stated that representing all parities with a fixed linear feature map requires exponentially many features. This problem proves that claim and shows that the bound is tight.

Suppose a fixed feature map $\varphi : \{-1,+1\}^d \to \mathbb{R}^D$ can represent every parity as an exact linear predictor: for every $I \subseteq [d]$, there exists $w_I \in \mathbb{R}^D$ such that

$$\forall x \in \{-1,+1\}^d, \quad \chi_I(x) = \langle w_I, \varphi(x) \rangle.$$

This is exact equality as real-valued functions, not merely equality after applying sign.

Prove that $D \geq 2^d$.

Hint: form the $2^d \times 2^d$ matrix $H$ whose rows are indexed by $I \subseteq [d]$, columns are indexed by $x \in \{-1,+1\}^d$, and entries are $H_{I,x} = \chi_I(x)$. Show that the rows of $H$ are orthogonal, hence $\mathrm{rank}(H) = 2^d$. Then use the factorization $H = W\Phi$ induced by the feature map.

Then prove the matching upper bound: define a feature map with one coordinate for each subset $I \subseteq [d]$, namely

$$\varphi(x) = (\chi_I(x))_{I \subseteq [d]}.$$

Show that every parity is represented exactly by a linear predictor in this feature space.

Explain how the lower and upper bounds connect to the Week 5 message: fixed-feature convex learning can be statistically and computationally clean, but the required feature dimension may be exponential.

**Solution.**

This is the same parity family as the lecture's $\{0,1\}^d$ convention, written in the equivalent $\{-1,+1\}^d$ product notation.

Define the matrix $H \in \mathbb{R}^{2^d \times 2^d}$ by

$$H_{I,x} = \chi_I(x),$$

with rows indexed by subsets $I \subseteq [d]$ and columns indexed by $x \in \{-1,+1\}^d$. For two subsets $I, J$,

$$\sum_{x \in \{-1,+1\}^d} H_{I,x} H_{J,x} = \sum_{x \in \{-1,+1\}^d} \chi_I(x)\chi_J(x) = \sum_{x \in \{-1,+1\}^d} \chi_{I \triangle J}(x),$$

where $I \triangle J$ is the symmetric difference.

If $I = J$, this sum is $2^d$. If $I \neq J$, choose some coordinate $\ell \in I \triangle J$. Pair each $x$ with the point obtained by flipping coordinate $\ell$. The values of $\chi_{I \triangle J}$ on the paired points have opposite signs, so the sum is 0. Therefore the rows of $H$ are nonzero and mutually orthogonal, and $\mathrm{rank}(H) = 2^d$.

Now suppose every parity is represented by the fixed feature map $\varphi : \{-1,+1\}^d \to \mathbb{R}^D$. Let $W$ be the $2^d \times D$ matrix whose row indexed by $I$ is $w_I^T$, and let $\Phi$ be the $D \times 2^d$ matrix whose column indexed by $x$ is $\varphi(x)$. The exact representation condition says

$$H = W\Phi.$$

Hence $2^d = \operatorname{rank}(H) \le \operatorname{rank}(W\Phi) \le D$. Thus $D \ge 2^d$.

For the matching upper bound, define

$$\varphi(x) = (\chi_J(x))_{J\subseteq[d]} \in \mathbb{R}^{2^d}.$$

To represent the parity $\chi_I$, choose $w_I$ to be the standard basis vector in the coordinate indexed by $I$. Then

$$\langle w_I, \varphi(x)\rangle = \chi_I(x)$$

for every $x$.

The conclusion is that a fixed feature map can make parity learning linear and convex only by including exponentially many features. The optimization over the top linear weights is clean once $\varphi$ is fixed, but the feature dimension needed to represent all parities is $2^d$.

4. **(8 points) Same predictors, different optimization geometry.**

   Let $S = ((x_1, y_1), ..., (x_n, y_n))$ with $x_i \in \mathbb{R}^d$ and $y_i \in \mathbb{R}$. Consider squared loss.

   The fixed-feature linear model is $f_\beta(x) = \langle \beta, x\rangle$ with empirical risk

   $$L_{\text{lin}}(\beta) = \frac{1}{n}\sum_{i=1}^{n} (\langle \beta, x_i\rangle - y_i)^2.$$

   The two-layer linear network with one hidden unit, with parameters $u \in \mathbb{R}^d$ and $v \in \mathbb{R}$, is

   $$f_{u,v}(x) = v\langle u, x\rangle$$

   with empirical risk

   $$L_{\text{net}}(u, v) = \frac{1}{n}\sum_{i=1}^{n} (v\langle u, x_i\rangle - y_i)^2.$$

   Prove the following:
   - the two parameterizations represent the same set of linear predictors;
   - $L_{\text{lin}}$ is convex in $\beta$;
   - $L_{\text{net}}$ is not necessarily convex in $(u, v)$ by giving a concrete dataset $S$ and a Jensen inequality violation;
   - if $\beta^*$ minimizes $L_{\text{lin}}$, then every factorization $\beta^* = vu$ gives a global minimizer of $L_{\text{net}}$.

   Explain what this example shows about replacing fixed features by learned features.

**Solution.**

The two-layer model satisfies

$$f_{u,v}(x) = v\langle u, x\rangle = \langle vu, x\rangle.$$

Thus it represents the linear predictor with coefficient vector $\beta = vu$. Conversely, every linear predictor $\beta$ is represented by taking $v = 1$ and $u = \beta$. Therefore the two parameterizations represent the same set of linear predictors.

The fixed-feature empirical risk is

$$L_{\text{lin}}(\beta) = \frac{1}{n}\sum_{i=1}^{n}(\langle \beta, x_i\rangle - y_i)^2.$$

Each summand is the square of an affine function of $\beta$, hence convex. Equivalently, its Hessian is positive semidefinite. Therefore $L_{\text{lin}}$ is convex in $\beta$.

The network objective need not be convex. Take $d = 1$ and one sample $(x_1, y_1) = (1, 1)$. Then

$$L_{\text{net}}(u, v) = (uv - 1)^2.$$

At $(u, v) = (1, 1)$ the loss is 0, and at $(u, v) = (-1, -1)$ the loss is also 0. Their midpoint is $(0, 0)$, where the loss is 1. Thus

$$L_{\text{net}}((0, 0)) = 1 > 0 = \frac{1}{2}L_{\text{net}}((1, 1)) + \frac{1}{2}L_{\text{net}}((-1, -1)),$$

violating Jensen's inequality.

Finally,

$$L_{\text{net}}(u, v) = L_{\text{lin}}(vu).$$

If $\beta^*$ minimizes $L_{\text{lin}}$ and $\beta^* = vu$, then

$$L_{\text{net}}(u, v) = L_{\text{lin}}(\beta^*) = \inf_{\beta} L_{\text{lin}}(\beta) \le L_{\text{lin}}(v'u') = L_{\text{net}}(u', v')$$

for every $(u', v')$. Hence every factorization of $\beta^*$ is a global minimizer of $L_{\text{net}}$.

This example shows that learning features can change the optimization geometry even when the represented predictors are unchanged. Fixed features give a convex problem in the linear weights; factoring the same linear weights into learned lower-layer and upper-layer parameters creates a nonconvex parameterization.