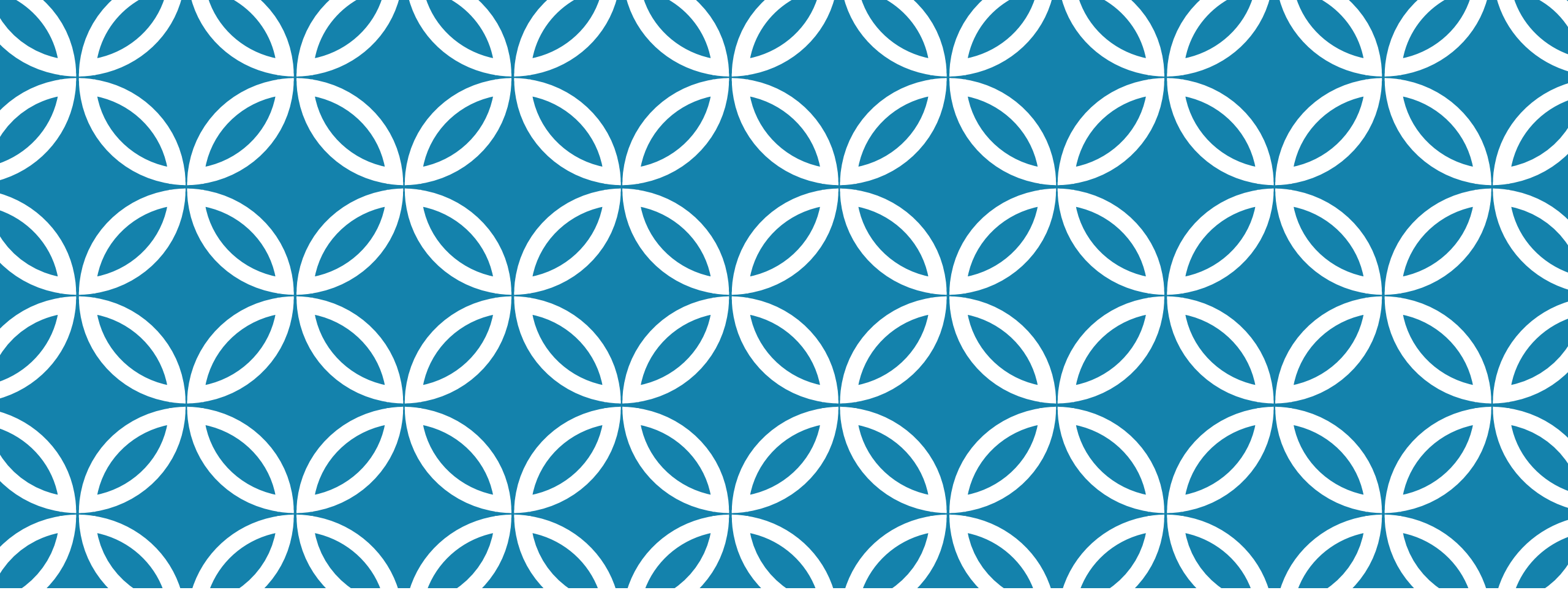
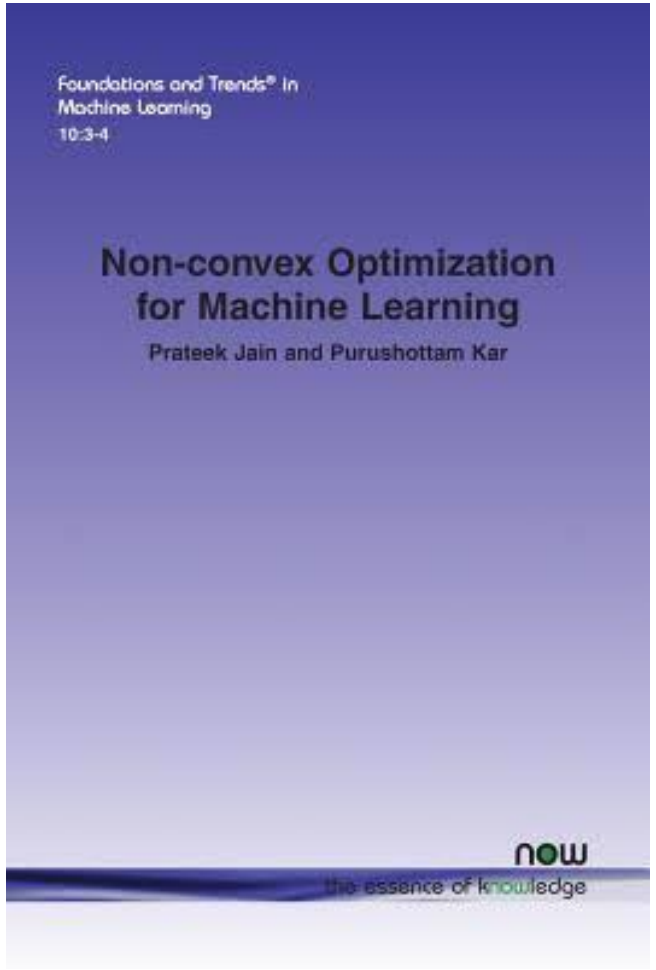


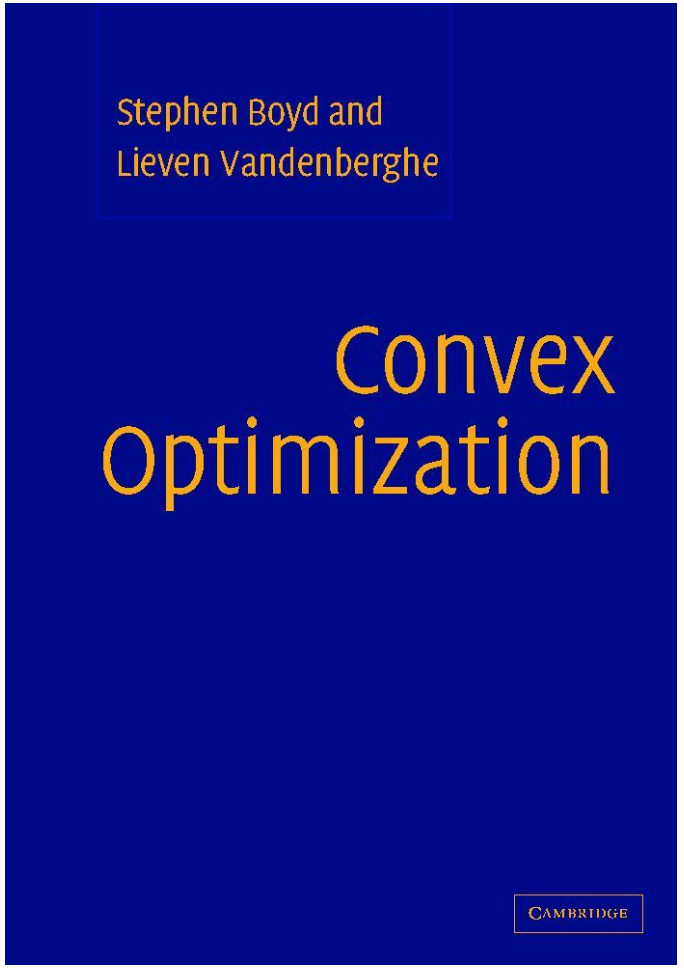
Non-convex Optimisation

Presenter: Bill

- **Mainly About**



https://arxiv.org/abs/1712.07897
Main book



https://web.stanford.edu/~boyd/cvxbook/
Reference

- **Mainly About**

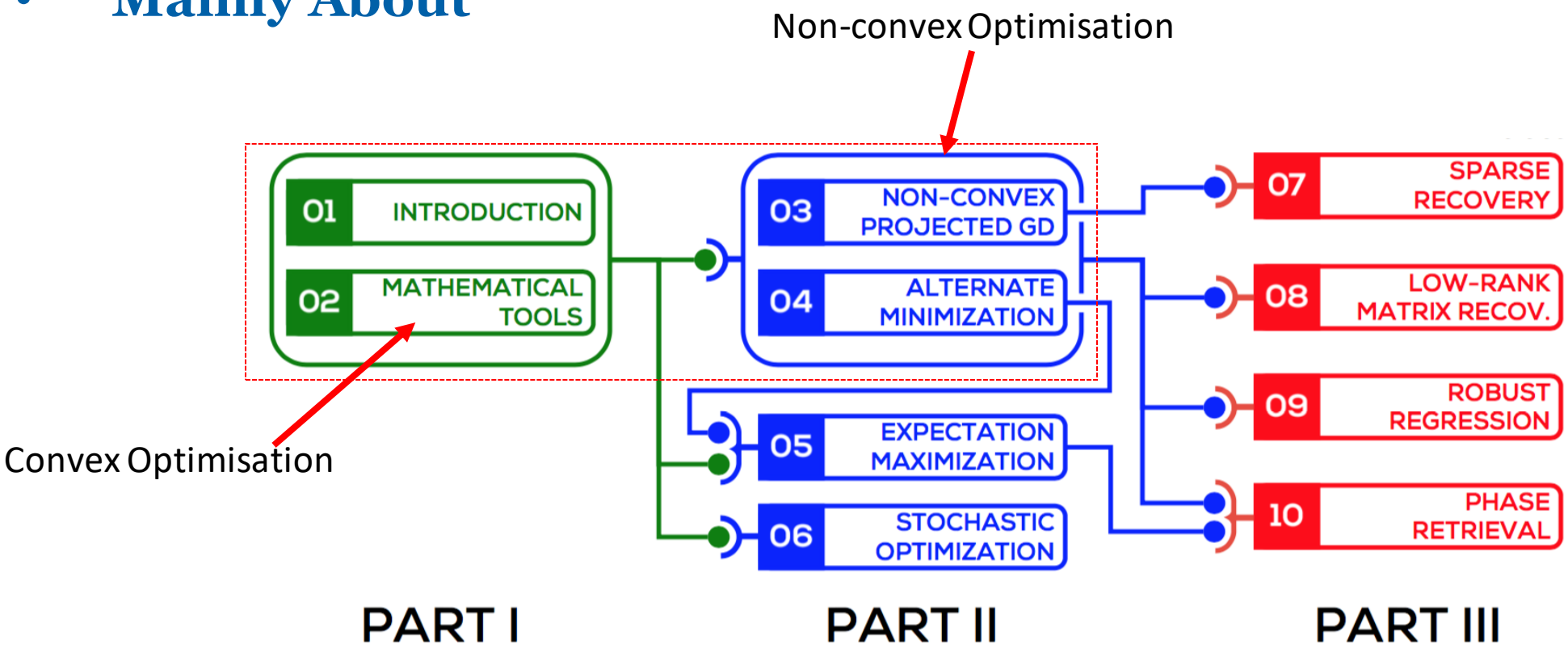


Figure 1: A schematic showing the suggested order of reading the sections. For example, concepts introduced in § 3 and 4 are helpful for § 9 but a thorough reading of § 6 is not required for the same. Similarly, we recommend reading § 5 after going through § 4 but a reader may choose to proceed to § 7 directly after reading § 3.

"Non-convex Optimization for Machine Learning"
First 1-4 Chapters in this presentation

Introduction

The generic form of an analytic optimization problem is the following

$$\min_{\mathbf{x}\in\mathbb{R}^p} f(\mathbf{x})$$
$$\text{s.t. } \mathbf{x} \in \mathcal{C},$$

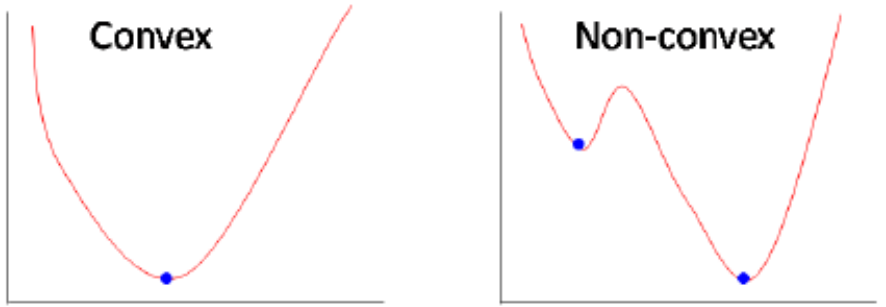
Subject to

Convex Optim Problem

where $\mathbf{x}$ is the *variable* of the problem, $f : \mathbb{R}^p \to \mathbb{R}$ is the *objective function* of the problem, and $\mathcal{C} \subseteq \mathbb{R}^p$ is the *constraint set* of the problem. When used in a machine learning setting, the objective function allows the algorithm designer to encode proper and expected behavior for the machine learning model, such as fitting well to training data with respect to some loss function, whereas the constraint allows restrictions on the model to be encoded, for instance, restrictions on model size.

In Boyd's Book

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq b_i, \quad i = 1, \ldots, m. \end{array}$$



• Non-convex Optim Examples – Sparse Regression

1.3. EXAMPLES OF NON-CONVEX OPTIMIZATION PROBLEMS

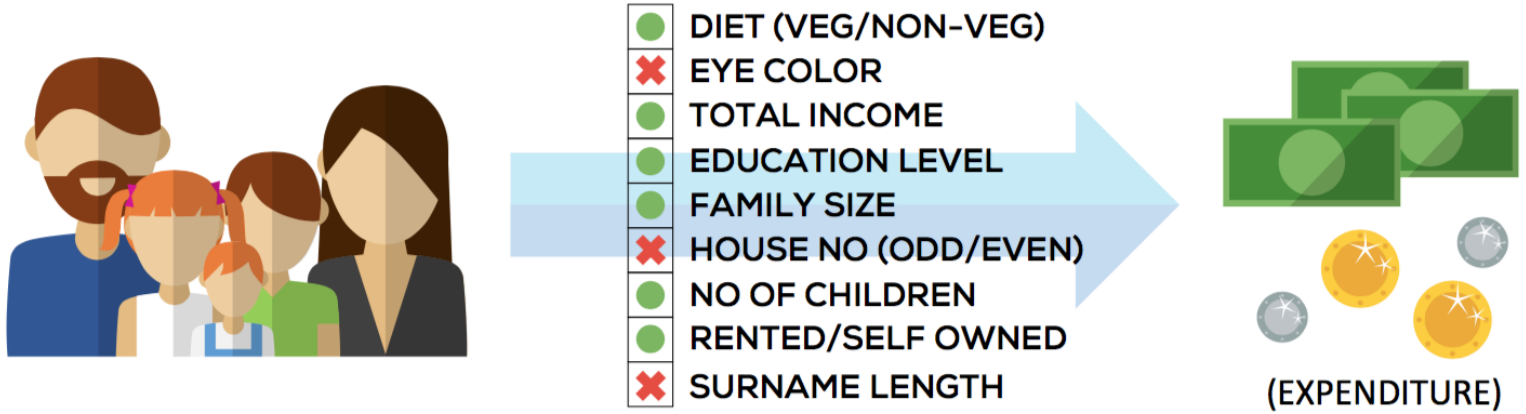


Figure 1.1: Not all available parameters and variables may be required for a prediction or learning task. Whereas the family size may significantly influence family expenditure, the eye color of family members does not directly or significantly influence it. Non-convex optimization techniques, such as sparse recovery, help discard irrelevant parameters and promote compact and accurate models.

A popular way to recover $\mathbf{w}^*$ is using the *least squares* formulation

$$\widehat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^p} \sum_{i=1}^{n} \left(y_i - \mathbf{x}_i^\top \mathbf{w}\right)^2.$$

The linear regression problem as well as the least squares estimator, are extremely well studied and their behavior, precisely known. However, this age-old problem acquires new dimensions in situations where, either we expect only a few of the p features/covariates to be actually relevant to the problem but do not know their identity, or else are working in extremely data-starved settings i.e., $n \ll p$.

(1) (2)

Discussed in Chapter 3 in detail

• Non-convex Optim Examples – Sparse Regression

Both these problems can be handled by the *sparse recovery* approach, which seeks to fit a sparse model vector (i.e., a vector with say, no more than s non-zero entries) to the data. The least squares formulation, modified as a sparse recovery problem, is given below

$$\widehat{\mathbf{w}}_{\text{sp}} = \arg\min_{\mathbf{w}\in\mathbb{R}^p} \sum_{i=1}^{n} \left(y_i - \mathbf{x}_i^\top \mathbf{w}\right)^2$$
$$\text{s.t. } \mathbf{w} \in \mathcal{B}_0(s),$$

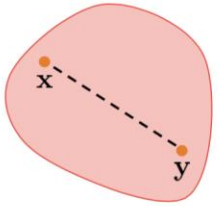
S-sparse

Although the objective function in the above formulation is convex, the constraint $\|\mathbf{w}\|_0 \leq s$ (equivalently $\mathbf{w} \in \mathcal{B}_0(s)$ – see list of mathematical notation at the beginning of this monograph) corresponds to a non-convex constraint set[1]. Sparse recovery effortlessly solves the twin problems of discarding irrelevant covariates and countering data-starvation since typically, only $n \geq s \log p$ (as opposed to $n \geq p$) data points are required for sparse recovery to work which drastically reduces the data requirement. Unfortunately however, sparse-recovery is an NP-hard problem [Natarajan, 1995].

- The support of a vector $\mathbf{x}$ is denoted by $\text{supp}(x) := \{i : \mathbf{x}_i \neq 0\}$. A vector x is referred to as s-sparse if $|\text{supp}(x)| \leq s$.

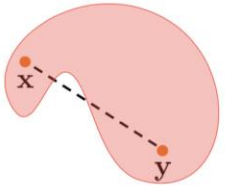
• Non-convex Optim Examples – Sparse Regression

Both these problems can be handled by the *sparse recovery* approach, which seeks to fit a sparse model vector (i.e., a vector with say, no more than s non-zero entries) to the data. The least squares formulation, modified as a sparse recovery problem, is given below



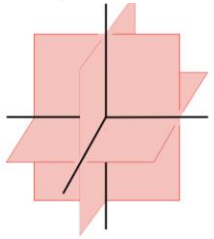
$\mathcal{C} \subseteq \mathbb{R}^d$

CONVEX SET



$\mathcal{C} \subseteq \mathbb{R}^d$

NON-CONVEX SET



$\mathcal{B}_0(2) \subseteq \mathbb{R}^3$

NON-CONVEX SET

$$\text{sp} = \arg\min_{\mathbf{w} \in \mathbb{R}^p} \sum_{i=1}^{n} \left(y_i - \mathbf{x}_i^\top \mathbf{w} \right)^2 \quad \text{s.t. } \mathbf{w} \in \mathcal{B}_0(s),$$

xi$_T$

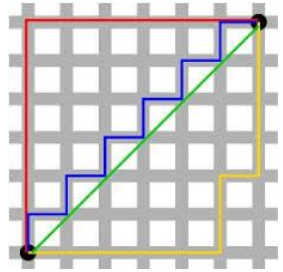
1*p

w

p*1

Although the objective function in the above formulation is convex, the constraint $\|\mathbf{w}\|_0 \leq s$ (equivalently $\mathbf{w} \in \mathcal{B}_0(s)$ – see list of mathematical notation at the beginning of this monograph) corresponds to a non-convex constraint set[1]. Sparse recovery effortlessly solves the twin problems of discarding irrelevant covariates and countering data-starvation since typically, only $n \geq s \log p$ (as opposed to $n \geq p$) data points are required for sparse recovery to work which drastically reduces the data requirement. Unfortunately however, sparse-recovery is an NP-hard problem [Natarajan, 1995].

- Balls with respect to various norms are denoted as $\mathcal{B}_q(r) := \left\{ \mathbf{x} \in \mathbb{R}^p, \|\mathbf{x}\|_q \leq r \right\}$. As a special case the notation $\mathcal{B}_0(s)$ is used to denote the set of s-sparse vectors.



- **Non-convex Optim Examples – low rank matrix recovery**

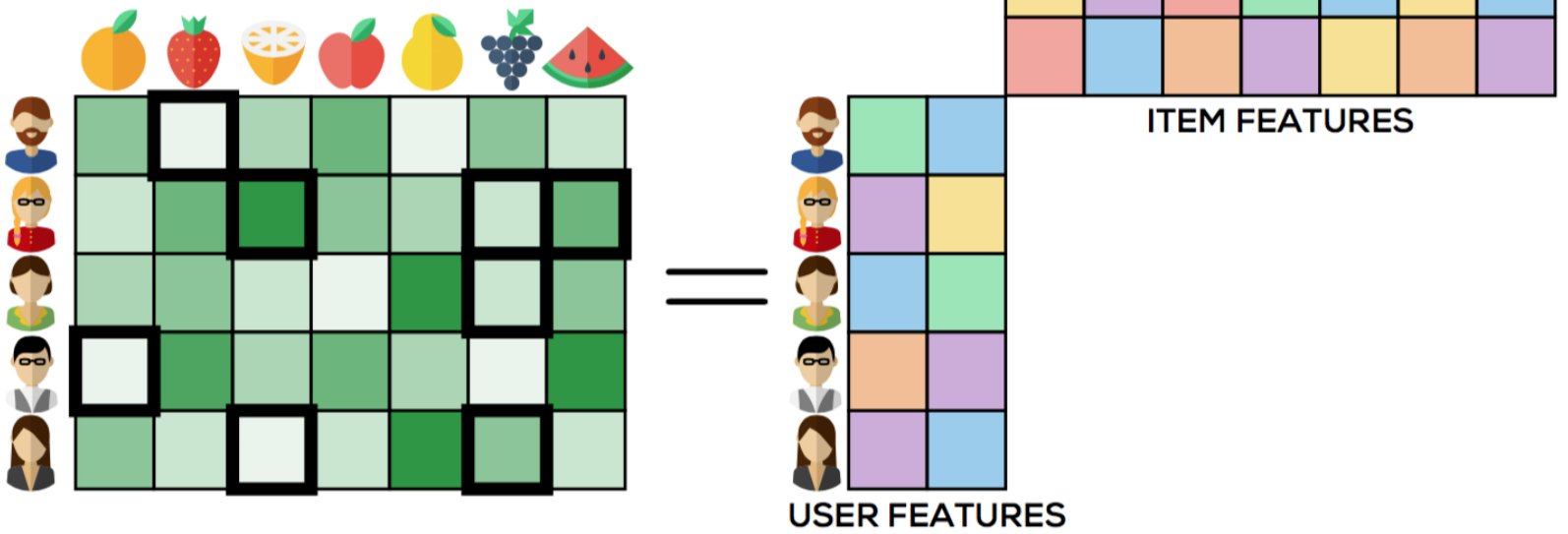


Figure 1.2: Only the entries of the ratings matrix with thick borders are observed. Notice that users rate infrequently and some items are not rated even once. Non-convex optimization techniques such as low-rank matrix completion can help recover the unobserved entries, as well as reveal hidden features that are descriptive of user and item properties, as shown on the right hand side.

m users $u_1,...,u_m$ and n items $a_1,...,a_n$, we have an $m \times n$ preference matrix $A = [A_{ij}]$ where A_{ij} encodes the preference of the i_{th} user for the j_{th} item.

- # Non-convex Optim Examples – low rank matrix recovery

If we denote by $\Omega \subset [m] \times [n]$, the set of observed entries of A, then the low rank matrix completion problem can be written as

$$\widehat{A}_{\text{lr}} = \underset{X \in \mathbb{R}^{m \times n}}{\arg\min} \sum_{(i,j) \in \Omega} (X_{ij} - A_{ij})^2$$
$$\text{s.t. } \text{rank}(X) \leq r,$$

This formulation also has a convex objective but a non-convex rank constraint[2]. This problem can be shown to be NP-hard as well. Interestingly, we can arrive at an alternate formulation by imposing the low-rank constraint indirectly. It turns out that[3] assuming the ratings matrix to have rank at most r is equivalent to assuming that the matrix A can be written as $A = UV^\top$ with the matrices $U \in \mathbb{R}^{m \times r}$ and $V \in \mathbb{R}^{n \times r}$ having at most r columns. This leads us to the following alternate formulation

$$\widehat{A}_{\text{lv}} = \underset{\substack{U \in \mathbb{R}^{m \times r} \\ V \in \mathbb{R}^{n \times r}}}{\arg\min} \sum_{(i,j) \in \Omega} \left(U_i^\top V_j - A_{ij} \right)^2.$$

There are no constraints in the formulation. However, the formulation requires joint optimization over a pair of variables (U, V) instead of a single variable. More importantly, it can be shown[4] that the objective function is non-convex in (U, V).

- **Diagonalisation and SVD**

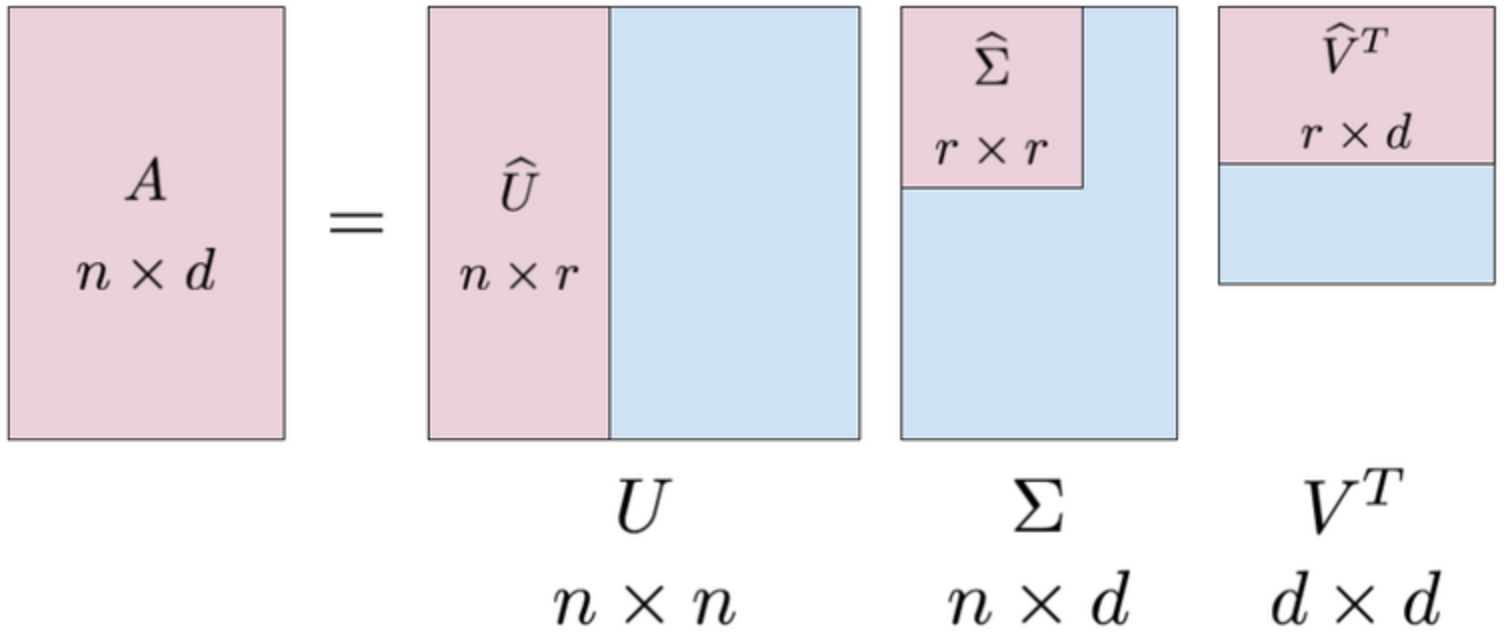
$$\mathrm{M} \rightarrow \begin{pmatrix} 2 & 0 & 1 \\ 1 & 1 & 1 \\ -2 & 0 & -1 \end{pmatrix}$$

$$M = P.D.P^{-1}$$

$$\mathrm{D} \rightarrow \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

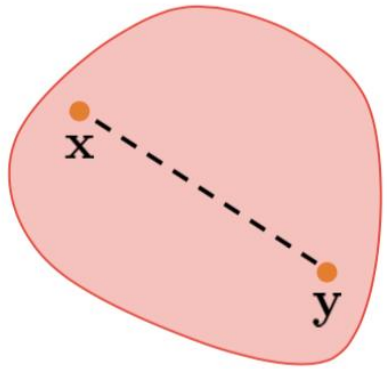
$$\mathrm{P} \rightarrow \begin{pmatrix} -\frac{1}{\sqrt{2}} & 0 & -\frac{1}{\sqrt{6}} \\ 0 & 1 & -\frac{1}{\sqrt{6}} \\ \frac{1}{\sqrt{2}} & 0 & \sqrt{\frac{2}{3}} \end{pmatrix}$$

Orthonormal



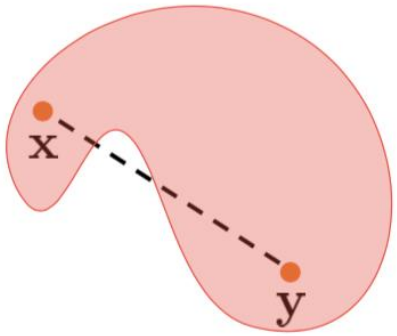
Rank(A) = Rank(∑)

- **Chapter 2: Convex Combination & Set & Function**



$\mathcal{C} \subseteq \mathbb{R}^d$

CONVEX SET



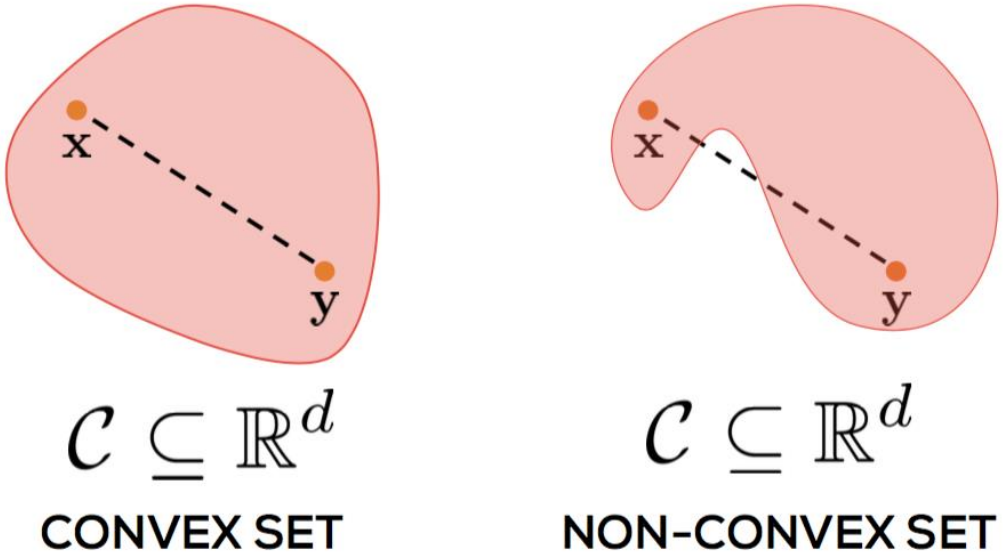
$\mathcal{C} \subseteq \mathbb{R}^d$

NON-CONVEX SET

A set that is closed under arbitrary convex combinations is a convex set. A standard definition is given below. Geometrically speaking, convex sets are those that contain all line segments that join two points inside the set. As a result, they cannot have any inward "bulges".

Definition 2.2 (Convex Set). *A set* $\mathcal{C} \in \mathbb{R}^p$ *is considered convex if, for every* $\mathbf{x}, \mathbf{y} \in \mathcal{C}$ *and* $\lambda \in [0, 1]$, *we have* $(1 - \lambda) \cdot \mathbf{x} + \lambda \cdot \mathbf{y} \in \mathcal{C}$ *as well.*

• Convex Combination & Set & Function

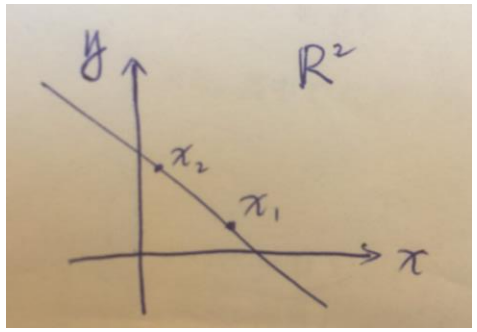


A set is convex if and only if it contains every convex combination of its points. A convex combination of points can be thought of as a mixture or weighted average of the points, with θ_i the fraction of x_i in the mixture.

We recall some basic definitions in convex analysis. Studying these will help us appreciate the structural properties of non-convex optimization problems later in the monograph. For the sake of simplicity, unless stated otherwise, we will assume that functions are continuously differentiable. We begin with the notion of a convex combination.

Definition 2.1 (Convex Combination). *A convex combination of a set of n vectors $\mathbf{x}_i \in \mathbb{R}^p$, $i = 1 \ldots n$ in an arbitrary real space is a vector $\mathbf{x}_{\boldsymbol{\theta}} := \sum_{i=1}^{n} \theta_i \mathbf{x}_i$ where $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_n)$, $\theta_i \geq 0$ and $\sum_{i=1}^{n} \theta_i = 1$.*

• Affine set & Convex set & Cone & Convex Hull



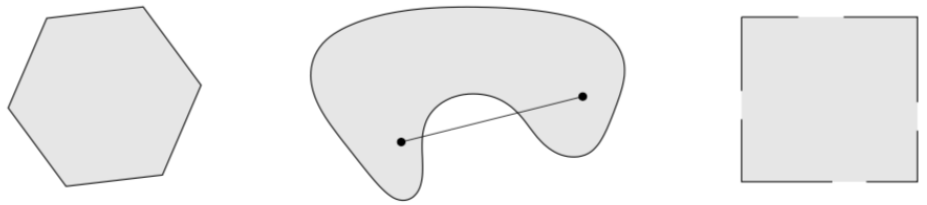
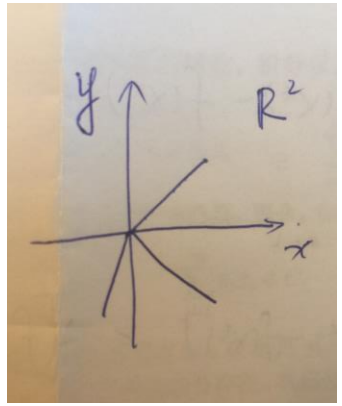
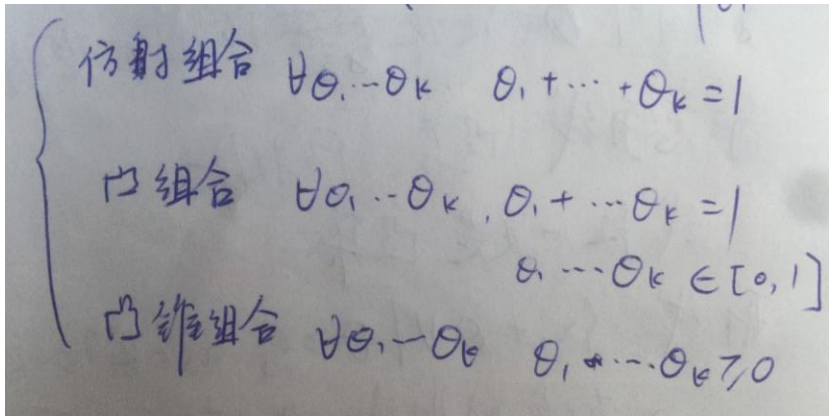
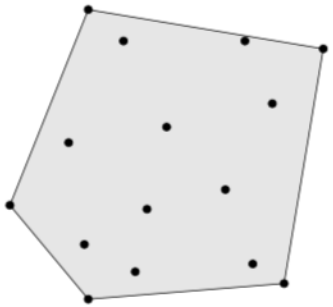


Figure 2.2 Some simple convex and nonconvex sets. *Left.* The hexagon, which includes its boundary (shown darker), is convex. *Middle.* The kidney shaped set is not convex, since the line segment between the two points in the set shown as dots is not contained in the set. *Right.* The square contains some boundary points but not others, and is not convex.







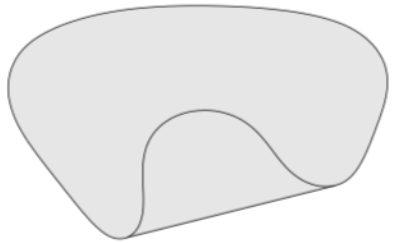
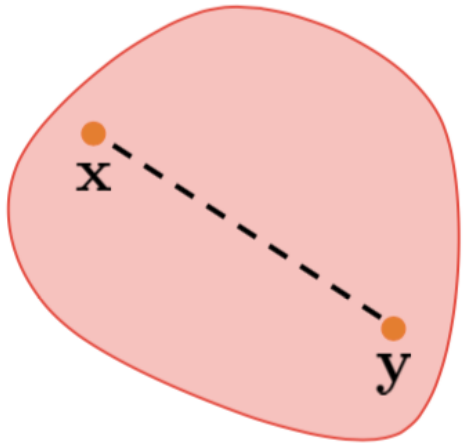


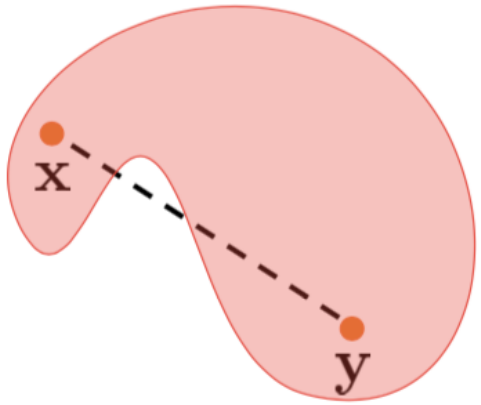
Figure 2.3 The convex hulls of two sets in $\mathbf{R}^2$. *Left.* The convex hull of a set of fifteen points (shown as dots) is the pentagon (shown shaded). *Right.* The convex hull of the kidney shaped set in figure 2.2 is the shaded set.

- **Convex Combination & Set & Function**



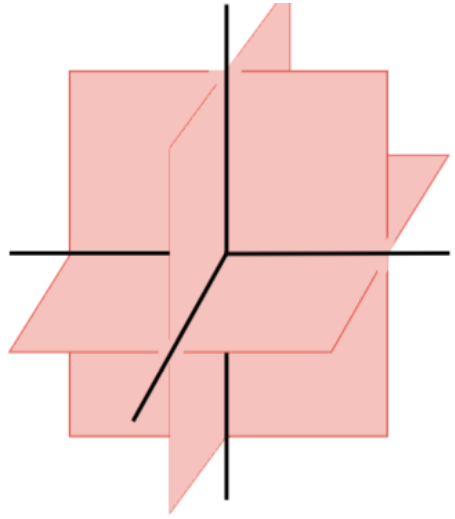
$\mathcal{C} \subseteq \mathbb{R}^d$

CONVEX SET



$\mathcal{C} \subseteq \mathbb{R}^d$

NON-CONVEX SET



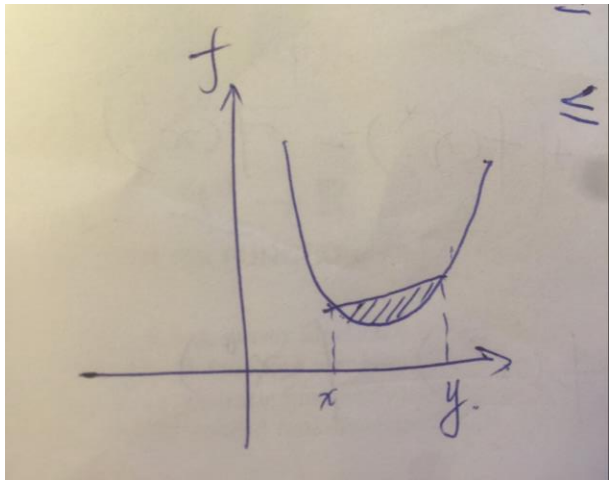
$\mathcal{B}_0(2) \subseteq \mathbb{R}^3$

NON-CONVEX SET

Figure 2.1: A convex set is closed under convex combinations. The presence of even a single uncontained convex combination makes a set non-convex. Thus, a convex set cannot have inward "bulges". In particular, the set of sparse vectors is non-convex.

- Balls with respect to various norms are denoted as $\mathcal{B}_q(r) := \left\{\mathbf{x} \in \mathbb{R}^p, \|\mathbf{x}\|_q \leq r\right\}$. As a special case the notation $\mathcal{B}_0(s)$ is used to denote the set of s-sparse vectors.

- # Convex Combination & Set & Function



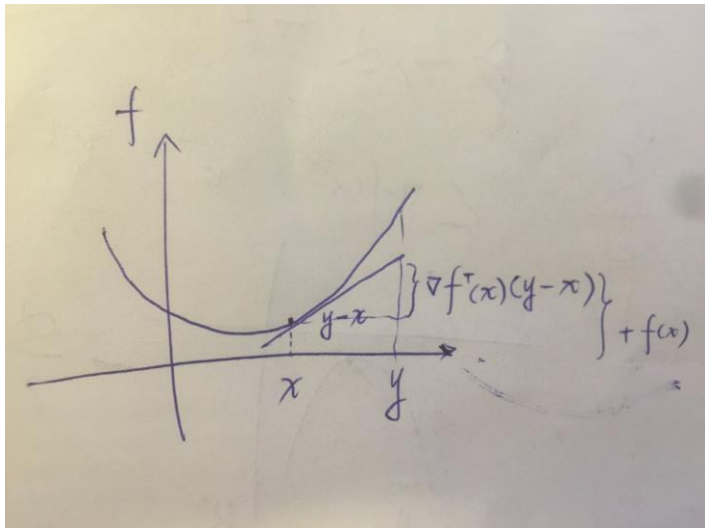
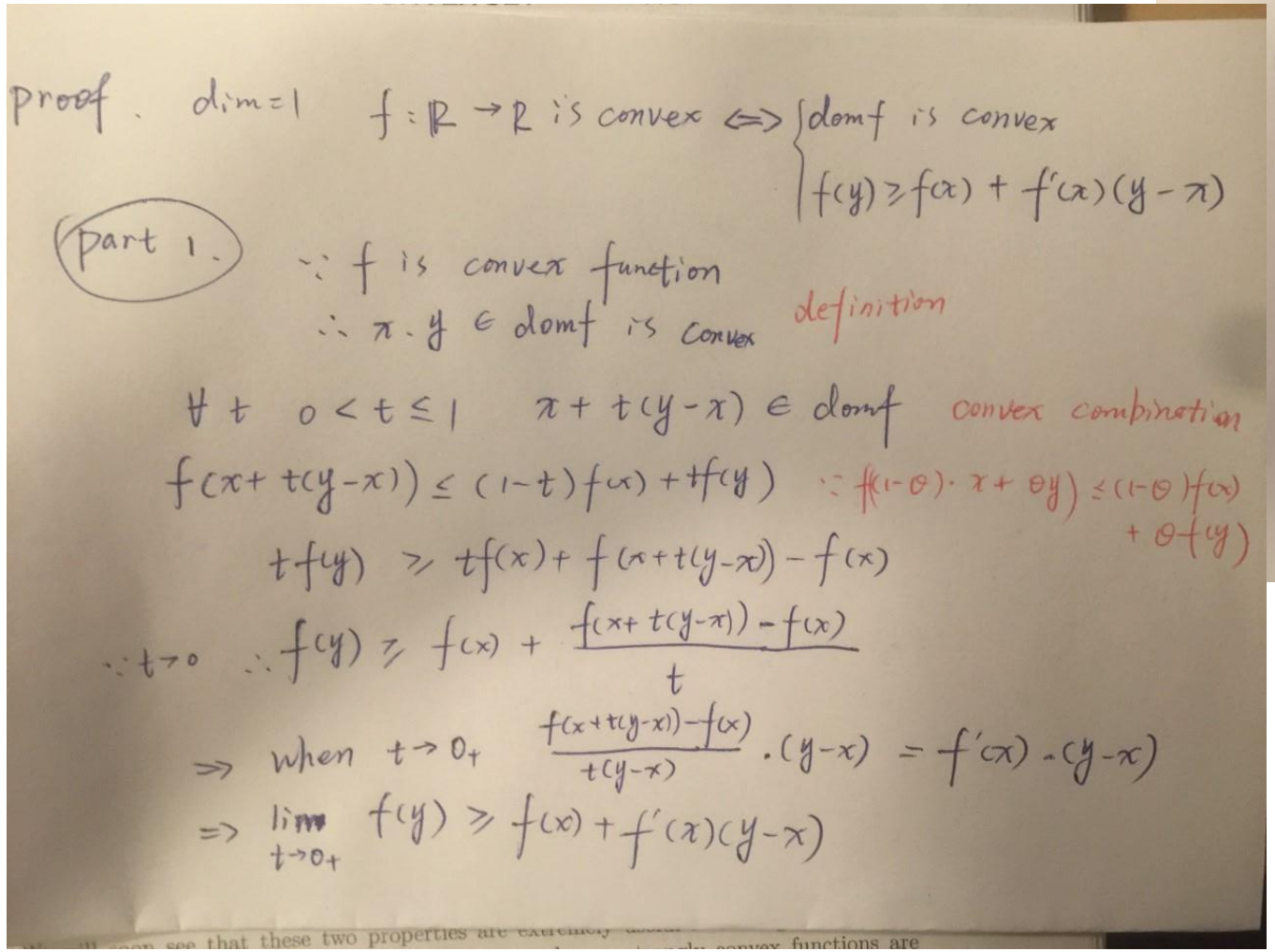


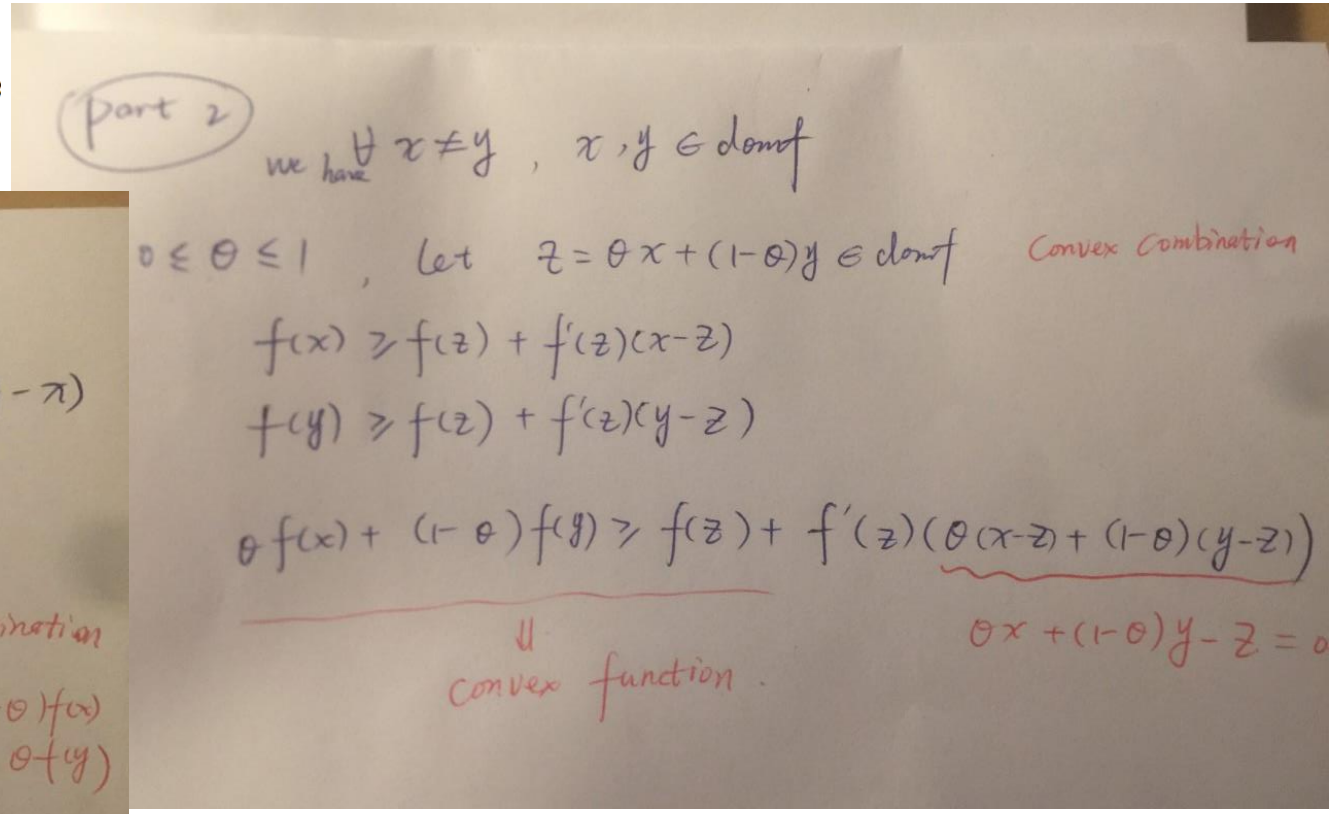
Figure 2.1 gives visual representations of prototypical convex and non-convex sets. A related notion is that of convex functions which have a unique behavior under convex combinations. There are several definitions of convex functions, those that are more basic and general, as well as those that are restrictive but easier to use. One of the simplest definitions of convex functions, one that does not involve notions of derivatives, defines convex functions $f : \mathbb{R}^p \to \mathbb{R}$ as those for which, for every $\mathbf{x}, \mathbf{y} \in \mathbb{R}^p$ and every $\lambda \in [0, 1]$, we have $f((1-\lambda)\cdot\mathbf{x}+\lambda\cdot\mathbf{y}) \leq (1-\lambda)\cdot f(\mathbf{x})+\lambda\cdot f(\mathbf{y})$. For continuously differentiable functions, a more usable definition follows.

Definition 2.3 (Convex Function)**.** *A continuously differentiable function* $f : \mathbb{R}^p \to \mathbb{R}$ *is considered convex if for every* $\mathbf{x}, \mathbf{y} \in \mathbb{R}^p$ *we have* $f(\mathbf{y}) \geq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle$*, where* $\nabla f(\mathbf{x})$ *is the gradient of* f *at* $\mathbf{x}$*.*

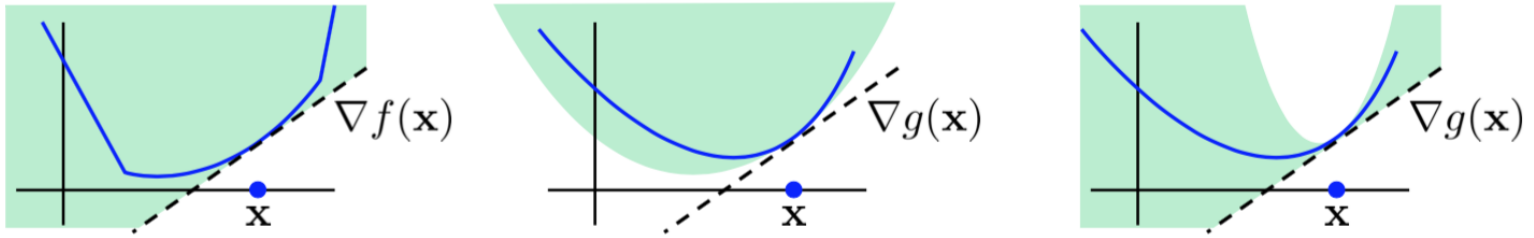
- **Convex Combination & Set & Function**

for every $\mathbf{x}, \mathbf{y} \in \mathbb{R}^p$ *we have* $f(\mathbf{y}) \geq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle$





- **Strongly Convex & Strongly Smooth**



$f : \mathbb{R}^d \to \mathbb{R}$ — CONVEX FUNCTION

$g : \mathbb{R}^d \to \mathbb{R}$ — STRONGLY CONVEX FUNCTION

$g : \mathbb{R}^d \to \mathbb{R}$ — STRONGLY SMOOTH CONVEX FUNCTION

May be non-convex

Figure 2.2: A convex function is lower bounded by its own tangent at all points. Strongly convex and smooth functions are, respectively, lower and upper bounded in the rate at which they may grow, by quadratic functions and cannot, again respectively, grow too slowly or too fast. In each figure, the shaded area describes regions the function curve is permitted to pass through.

Definition 2.4 (Strongly Convex/Smooth Function). *A continuously differentiable function $f : \mathbb{R}^p \to \mathbb{R}$ is considered α-strongly convex (SC) and β-strongly smooth (SS) if for every $\mathbf{x}, \mathbf{y} \in \mathbb{R}^p$, we have*

$$\frac{\alpha}{2}\|\mathbf{x}-\mathbf{y}\|_2^2 \leq f(\mathbf{y}) - f(\mathbf{x}) - \langle \nabla f(\mathbf{x}), \mathbf{y}-\mathbf{x}\rangle \leq \frac{\beta}{2}\|\mathbf{x}-\mathbf{y}\|_2^2 .$$

Definition 2.5 (Lipschitz Function). *A function $f : \mathbb{R}^p \to \mathbb{R}$ is B-Lipschitz if for every $\mathbf{x}, \mathbf{y} \in \mathbb{R}^p$, we have*

$$|f(\mathbf{x}) - f(\mathbf{y})| \leq B \cdot \|\mathbf{x}-\mathbf{y}\|_2 .$$

• Convex Projection

The projected gradient descent technique is a popular method for constrained optimization problems, both convex as well as non-convex. The *projection* step plays an important role in this technique. Given any closed set $\mathcal{C} \subset \mathbb{R}^p$, the projection operator $\Pi_{\mathcal{C}}(\cdot)$ is defined as

$$\Pi_{\mathcal{C}}(\mathbf{z}) := \arg\min_{\mathbf{x} \in \mathcal{C}} \|\mathbf{x} - \mathbf{z}\|_2 .$$

For instance, if $\mathcal{C} = \mathcal{B}_2(1)$ i.e., the unit L_2 ball, then projection is equivalent[3] to a normalization step

$$\Pi_{\mathcal{B}_2(1)}(\mathbf{z}) = \begin{cases} \mathbf{z}/\|\mathbf{z}\|_2 & \text{if } \|\mathbf{z}\| > 1 \\ \mathbf{z} & \text{otherwise} \end{cases} .$$

- Balls with respect to various norms are denoted as $\mathcal{B}_q(r) := \left\{\mathbf{x} \in \mathbb{R}^p, \|\mathbf{x}\|_q \leq r\right\}$. As a special case the notation $\mathcal{B}_0(s)$ is used to denote the set of s-sparse vectors.

- **Convex Projection**

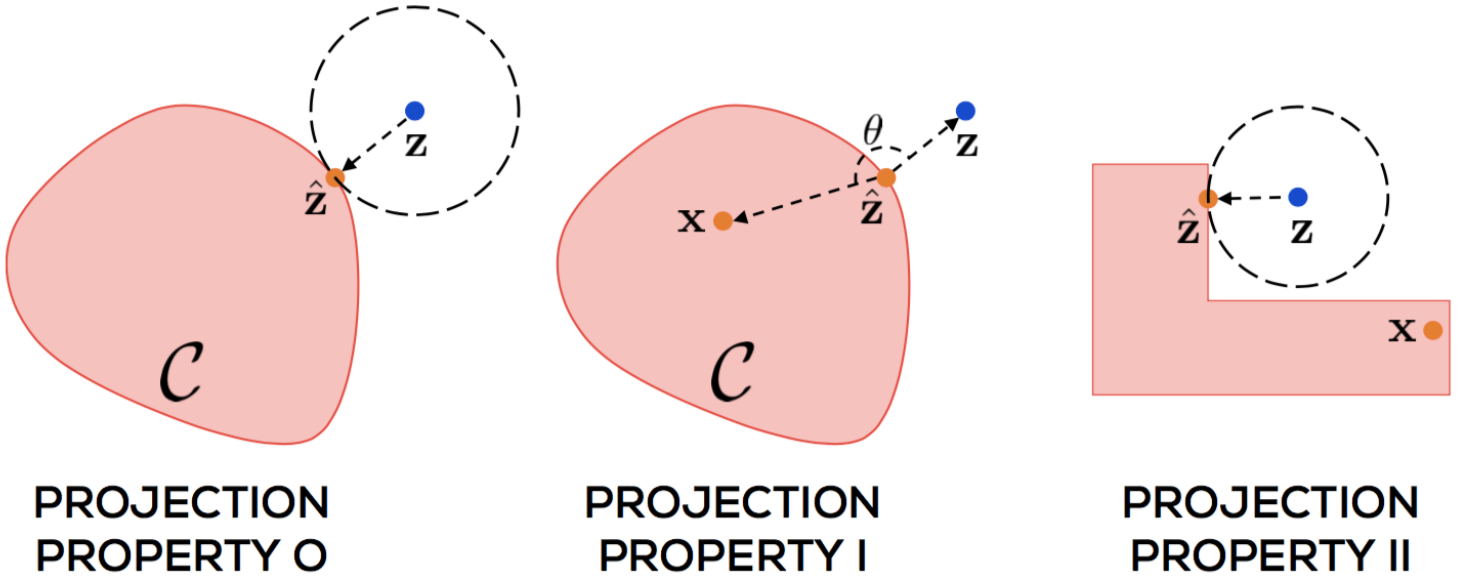


Figure 2.3: A depiction of projection operators and their properties. Projections reveal a closest point in the set being projected onto. For convex sets, projection property I ensures that the angle θ is always non-acute. Sets that satisfy projection property I also satisfy projection property II. Projection property II may be violated by non-convex sets. Projecting onto them may take the projected point $\mathbf{z}$ closer to certain points in the set (for example, $\hat{\mathbf{z}}$) but farther from others (for example, $\mathbf{x}$).

Lemma 2.2 (Projection Property-O). *For any set (convex or not) $\mathcal{C} \subset \mathbb{R}^p$ and $\mathbf{z} \in \mathbb{R}^p$, let $\hat{\mathbf{z}} := \Pi_{\mathcal{C}}(\mathbf{z})$. Then for all $\mathbf{x} \in \mathcal{C}$, $\|\hat{\mathbf{z}} - \mathbf{z}\|_2 \leq \|\mathbf{x} - \mathbf{z}\|_2$.*

This property follows by simply observing that the projection step solves the the optimization problem $\min_{\mathbf{x} \in \mathcal{C}} \|\mathbf{x} - \mathbf{z}\|_2$. Note that this property holds for all sets, whether convex or not. However, the following two properties necessarily hold only for convex sets.

- # Convex Projection

2.2. CONVEX PROJECTIONS

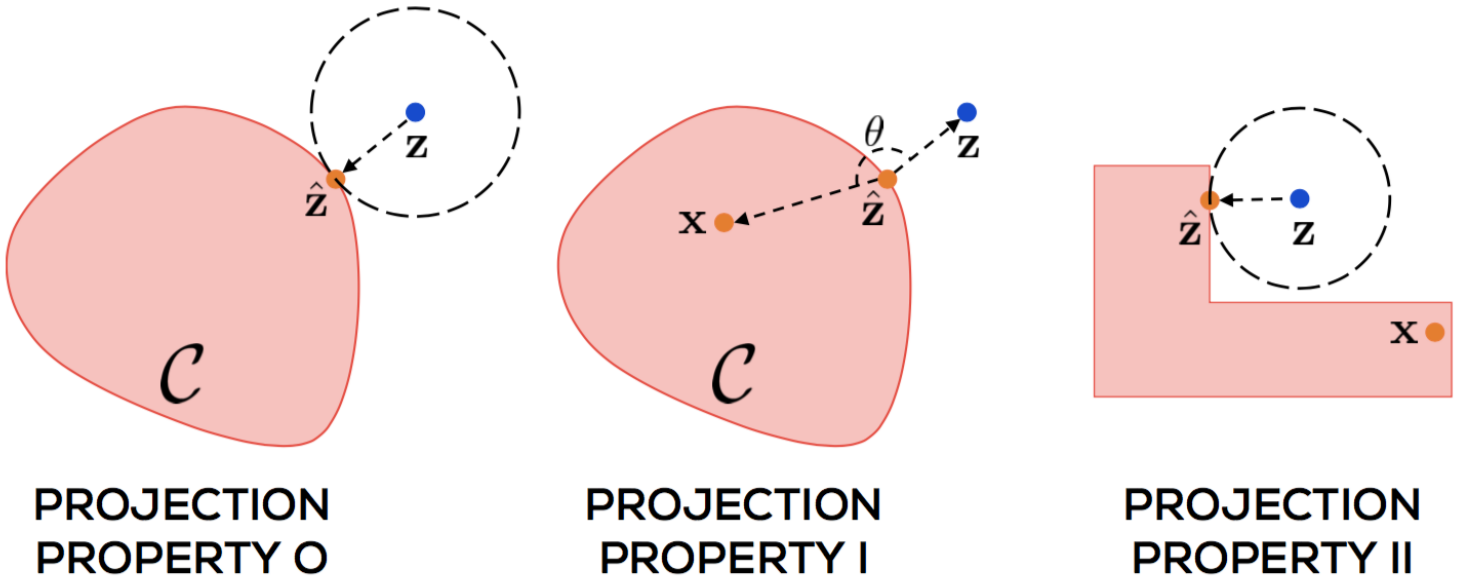
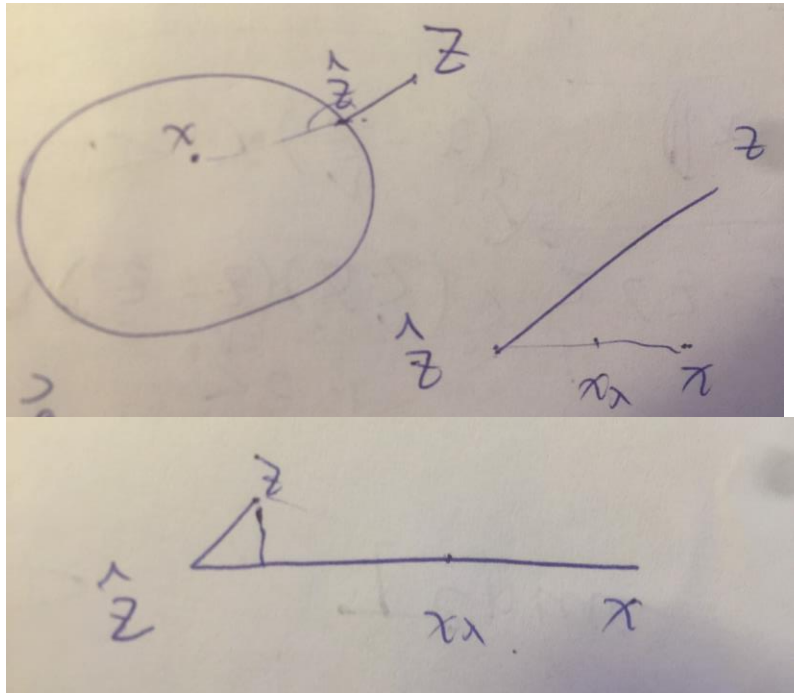


Figure 2.3: A depiction of projection operators and their properties. Projections reveal a closest point in the set being projected onto. For convex sets, projection property I ensures that the angle θ is always non-acute. Sets that satisfy projection property I also satisfy projection property II. Projection property II may be violated by non-convex sets. Projecting onto them may take the projected point $\mathbf{z}$ closer to certain points in the set (for example, $\widehat{\mathbf{z}}$) but farther from others (for example, $\mathbf{x}$).

Lemma 2.3 (Projection Property-I). *For any convex set* $\mathcal{C} \subset \mathbb{R}^p$ *and any* $\mathbf{z} \in \mathbb{R}^p$, *let* $\widehat{\mathbf{z}} := \Pi_{\mathcal{C}}(\mathbf{z})$. *Then for all* $\mathbf{x} \in \mathcal{C}$, $\langle \mathbf{x} - \widehat{\mathbf{z}}, \mathbf{z} - \widehat{\mathbf{z}} \rangle \leq 0$.

- **Convex Projection**



Proof. Contra-positive

assume $\exists\ x \in C$, we have $\langle x-\hat{z}, z-\hat{z}\rangle > 0$

$\hat{z}$, $x \in$ convex set C, $\forall\ \lambda \in [0,1]$, we have

$x_\lambda = \lambda \cdot x + (1-\lambda)\cdot \hat{z} \in C$

$\|z - x_\lambda\|_2 = \|z - \lambda x - (1-\lambda)\cdot \hat{z}\|_2$

$= \|z - \hat{z} - \lambda(x-\hat{z})\|_2$

$(z-x_\lambda)^2 = (z-\hat{z})^2 + \underbrace{[\lambda(\hat{z}-x)]^2}_{\geq 0} - 2\lambda(z-\hat{z})(x-\hat{z})$

If we let $2\lambda(z-\hat{z})(x-\hat{z}) > [\lambda(\hat{z}-x)]^2$

$\Rightarrow 2\lambda\langle z-\hat{z}, \hat{z}-x\rangle < -[\lambda(\hat{z}-x)]^2$

$\because \lambda \neq 0$

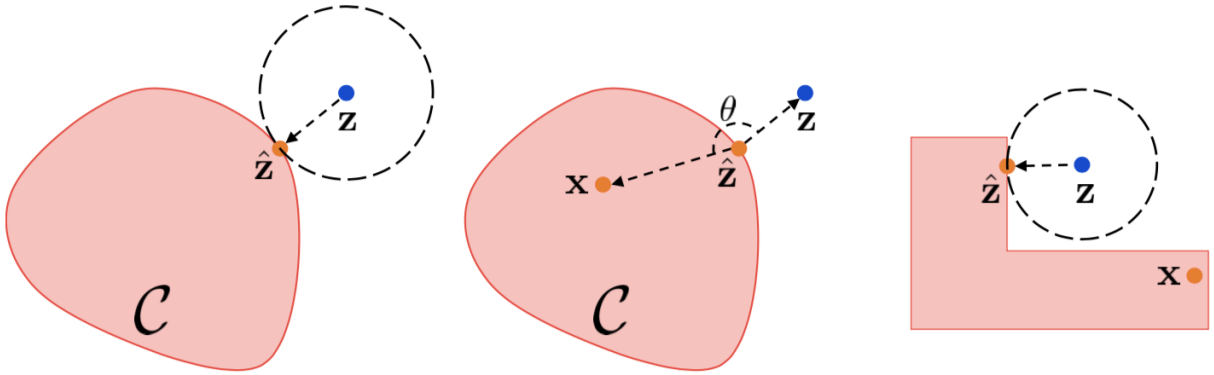
$\therefore 2\langle z-\hat{z}, \hat{z}-x\rangle < -\lambda(\hat{z}-x)^2$

$\Rightarrow \lambda < \dfrac{2\langle z-\hat{z}, x-z\rangle}{\|x-\hat{z}\|_2^2}$ >0

$\Rightarrow \exists\ \lambda$ let $\|z-x_\lambda\|_2 < \|z-\hat{z}\|_2$

in the convex set to $\mathbf{z}$ and prove the lemma. All that remains to be done is to find such a value of λ. The reader can verify that any value of $0 < \lambda < \min\left\{1, \frac{2\langle \mathbf{x}-\widehat{\mathbf{z}}, \mathbf{z}-\widehat{\mathbf{z}}\rangle}{\|\mathbf{x}-\widehat{\mathbf{z}}\|_2^2}\right\}$ suffices. Since we assumed $\langle \mathbf{x}-\widehat{\mathbf{z}}, \mathbf{z}-\widehat{\mathbf{z}}\rangle > 0$, any value of λ chosen this way is always in $(0,1]$. □

- **Convex Projection**



PROJECTION PROPERTY O | PROJECTION PROPERTY I | PROJECTION PROPERTY II

Lemma 2.4 (Projection Property-II). *For any convex set* $\mathcal{C} \subset \mathbb{R}^p$ *and any* $\mathbf{z} \in \mathbb{R}^p$, *let* $\widehat{\mathbf{z}} := \Pi_{\mathcal{C}}(\mathbf{z})$. *Then for all* $\mathbf{x} \in \mathcal{C}$, $\|\widehat{\mathbf{z}} - \mathbf{x}\|_2 \leq \|\mathbf{z} - \mathbf{x}\|_2$.

Proof. We have the following elementary inequalities

$$\begin{aligned}
\|\mathbf{z} - \mathbf{x}\|_2^2 &= \|(\widehat{\mathbf{z}} - \mathbf{x}) - (\widehat{\mathbf{z}} - \mathbf{z})\|_2^2 \\
&= \|\widehat{\mathbf{z}} - \mathbf{x}\|_2^2 + \|\widehat{\mathbf{z}} - \mathbf{z}\|_2^2 - 2\langle \widehat{\mathbf{z}} - \mathbf{x}, \widehat{\mathbf{z}} - \mathbf{z} \rangle \\
&\geq \|\widehat{\mathbf{z}} - \mathbf{x}\|_2^2 + \|\widehat{\mathbf{z}} - \mathbf{z}\|_2^2 \qquad \text{(Projection Property-I)} \\
&\geq \|\widehat{\mathbf{z}} - \mathbf{x}\|_2^2
\end{aligned}$$

□

Note that Projection Properties-I and II are also called *first order* properties and can be violated if the underlying set is non-convex. However, Projection Property-O, often called a *zeroth order* property, always holds, whether the underlying set is convex or not.

• Gradient Descent

Optimisation Algorithms:

- Without constraint (Gradient Descent Algorithm)
- With constraint (Projected Gradient Descent Algorithm)

The optimisation Algorithms we discussed are all iterative algorithms

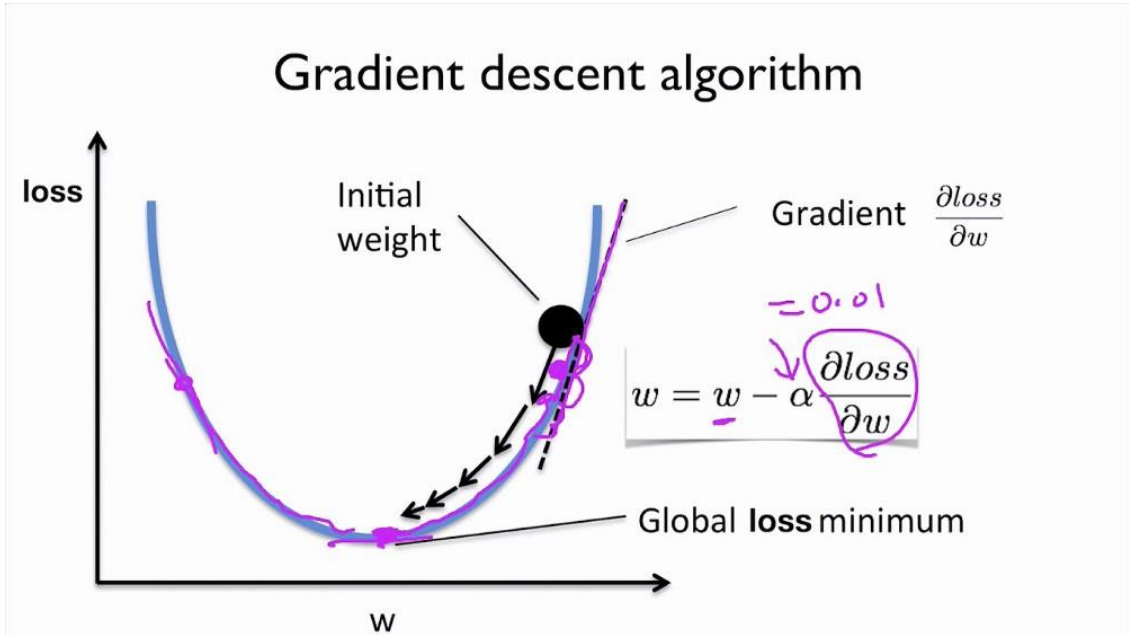
We now move on to study the projected gradient descent algorithm. This is an extremely simple and efficient technique that can effortlessly scale to large problems. Although we will apply this technique to non-convex optimization tasks later, we first look at its behavior on convex optimization problems as a warm up exercise. We warn the reader that the proof techniques used in the convex case do not apply directly to non-convex problems. Consider the following optimization problem:

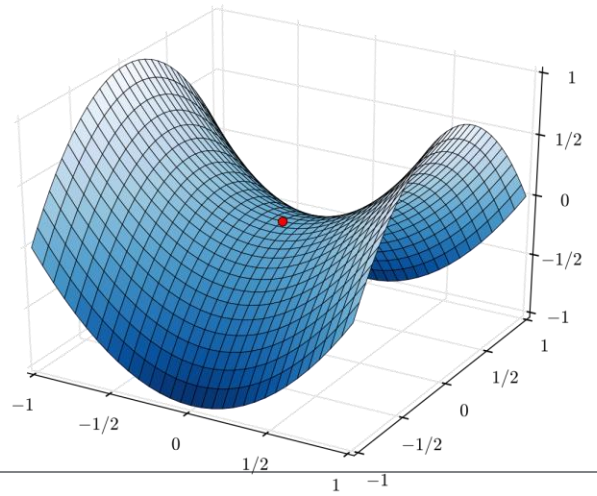
$$x_{t+1} \leftarrow x_t - \eta_t \cdot \nabla f(x_t)$$

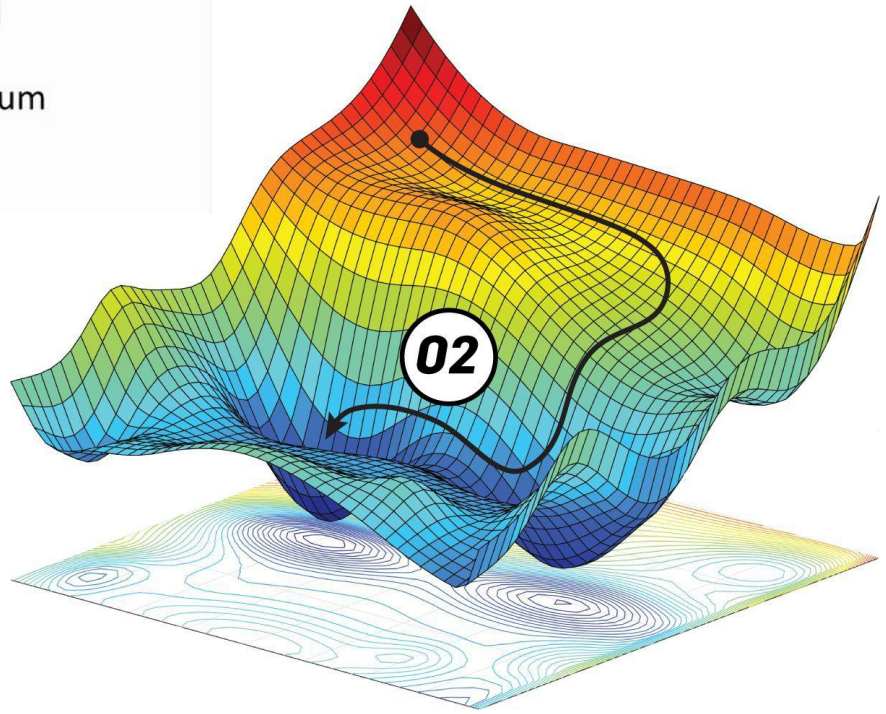
$$\min_{\mathbf{x} \in \mathbb{R}^p} f(\mathbf{x}) \quad \text{s.t. } \mathbf{x} \in \mathcal{C}. \qquad \text{(CVX-OPT)}$$

In the above optimization problem, $\mathcal{C} \subset \mathbb{R}^p$ is a convex constraint set and $f : \mathbb{R}^p \to \mathbb{R}$ is a convex objective function. We will assume that we have oracle access to the gradient and projection operators, i.e., for any point $\mathbf{x} \in \mathbb{R}^p$ we are able to access $\nabla f(\mathbf{x})$ and $\Pi_{\mathcal{C}}(\mathbf{x})$.

- # How do we train Deep Nets?







- **Gradient Descent**

$\nabla f(x) \approx 0$ $\begin{cases} f(x) \to f(x^*) \\ x \to x^* \end{cases}$

In much of this chapter (with the exception of §9.6) we assume that the objective function is *strongly convex* on S, which means that there exists an $m > 0$ such that

$$\nabla^2 f(x) \succeq mI \tag{9.7}$$

$$f(y) \geq f(x) + \nabla f(x)^T (y - x) + \frac{m}{2}\|y - x\|_2^2$$

which is the suboptimality of the point x, in terms of $\|\nabla f(x)\|_2$. The righthand side of (9.8) is a convex quadratic function of y (for fixed x). Setting the gradient with respect to y equal to zero, we find that $\tilde{y} = x - (1/m)\nabla f(x)$ minimizes the righthand side. Therefore we have

$$\begin{aligned} f(y) &\geq f(x) + \nabla f(x)^T (y - x) + \frac{m}{2}\|y - x\|_2^2 \\ &\geq f(x) + \nabla f(x)^T (\tilde{y} - x) + \frac{m}{2}\|\tilde{y} - x\|_2^2 \\ &= f(x) - \frac{1}{2m}\|\nabla f(x)\|_2^2. \end{aligned}$$

$\nabla f^T(x) + m(\tilde{y} - x) = 0 \quad \tilde{y} = x - \frac{\nabla f^T(x)}{m}$

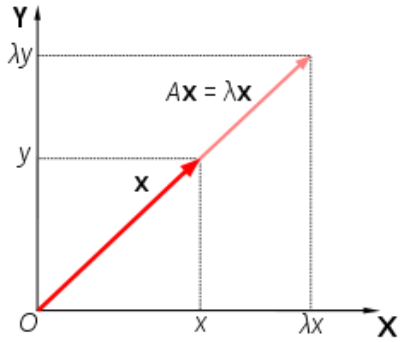
Since this holds for any $y \in S$, we have

$$p^\star \geq f(x) - \frac{1}{2m}\|\nabla f(x)\|_2^2.$$

$\|f(x) - f(x^*)\|_2 \leq \frac{1}{2m}\|\nabla f(x)\|_2^2$

- **Saddle Points**

$$f(x) = f(x_0) + (x - x_0)^T \nabla f(x_0) + \frac{1}{2}(x - x_0)^T H (x - x_0) + \cdots\cdots$$



① H is positive definite ⇒ ∀ eigenvalue > 0

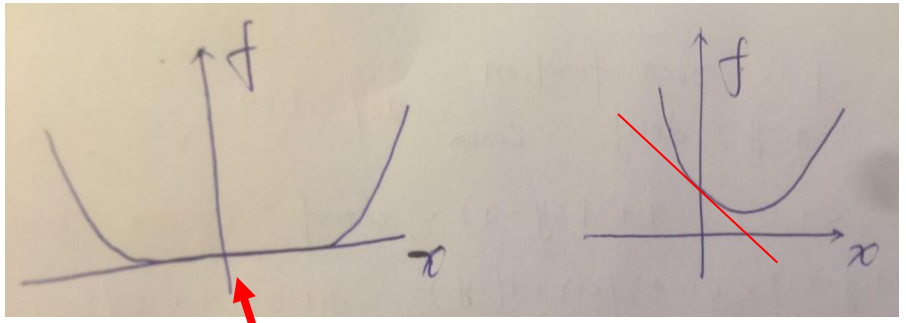
at point x_0 ⇒ local minima.

② H is negtive definite ⇒ ∀ eigenvalue < 0

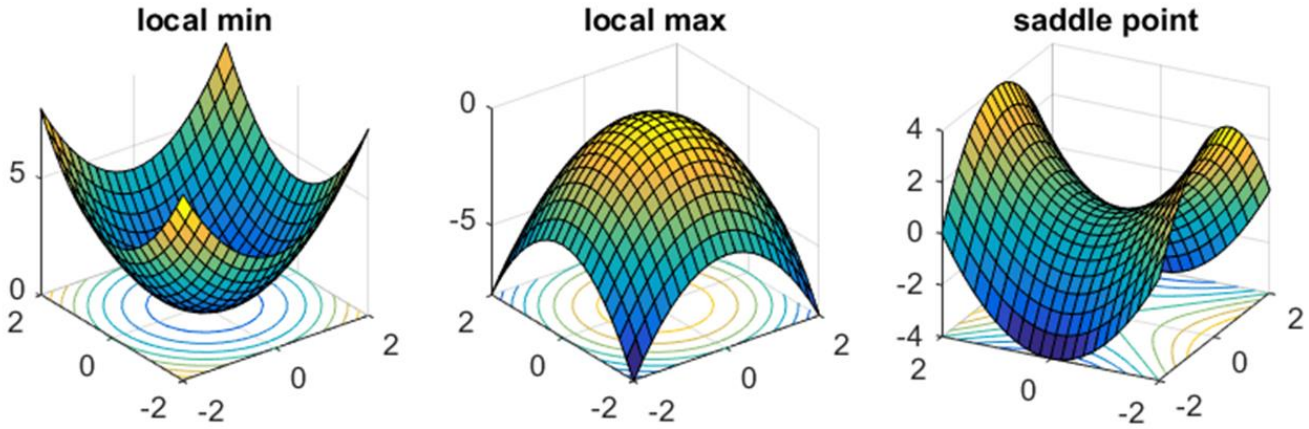
⇒ local maxima

③ some eigenvalue < 0, some > 0

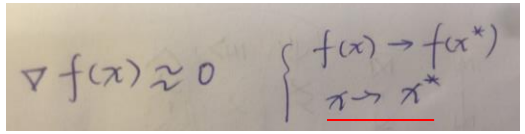
⇒ Saddle point



$H\lambda = 0$



- **Gradient Descent**

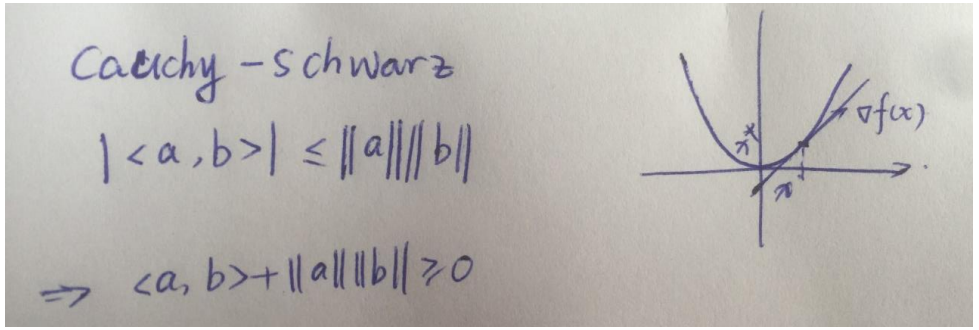


We can also derive a bound on $\|x - x^\star\|_2$, the distance between x and any optimal point $x^\star$, in terms of $\|\nabla f(x)\|_2$:

$$\|x - x^\star\|_2 \leq \frac{2}{m}\|\nabla f(x)\|_2. \tag{9.11}$$

To see this, we apply (9.8) with $y = x^\star$ to obtain

$$\begin{aligned} p^\star = f(x^\star) &\geq f(x) + \nabla f(x)^T(x^\star - x) + \frac{m}{2}\|x^\star - x\|_2^2 \\ &\geq f(x) - \|\nabla f(x)\|_2\|x^\star - x\|_2 + \frac{m}{2}\|x^\star - x\|_2^2, \end{aligned}$$



where we use the Cauchy-Schwarz inequality in the second inequality. Since $p^\star \leq f(x)$, we must have

$$-\|\nabla f(x)\|_2\,\|x^\star - x\|_2 + \frac{m}{2}\|x^\star - x\|_2^2 \leq 0,$$

- **Gradient Descent**

$\nabla f(x) \approx 0 \quad \begin{cases} f(x) \to f(x^*) \\ x \to x^* \end{cases}$

We can also derive a bound on $\|x - x^\star\|_2$, the distance between x and any optimal point $x^\star$, in terms of $\|\nabla f(x)\|_2$:

$$\|x - x^\star\|_2 \leq \frac{2}{m}\|\nabla f(x)\|_2. \tag{9.11}$$

$$\|f(x) - f(x^*)\|_2 \leq \frac{1}{2m}\|\nabla f(x)\|_2^2$$

Analyse: ① If m is large $\Rightarrow$ strongly convex

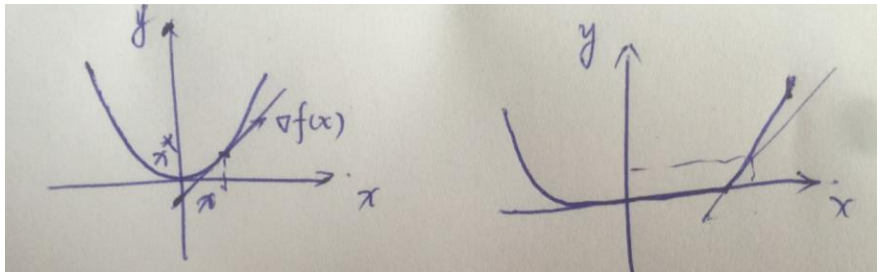
and $\nabla f(x) \to 0 \Rightarrow f(x) \to f^*(x)$, $x \to x^*$

② but if m is small.

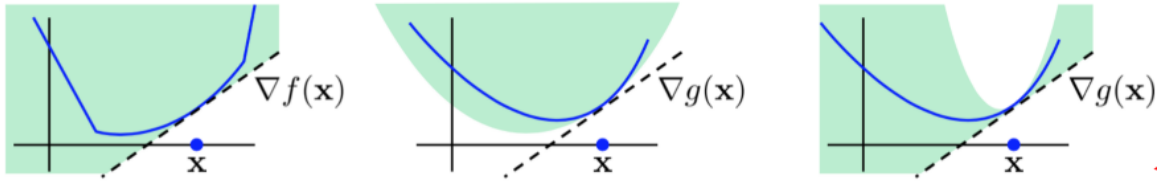
$\Rightarrow$ cannot guarantee above

③ when m is fixed, $\|f(x) - f^*(x)\|$ has square of $\nabla f(x)$

$\Rightarrow$ tighter bound $\Rightarrow \nabla f(x) \to 0$, $\|f(x) - f^*(x)\|$ improves more



- **Gradient Descent**



$-f : \mathbb{R}^d \to \mathbb{R}$ CONVEX FUNCTION

$-g : \mathbb{R}^d \to \mathbb{R}$ STRONGLY CONVEX FUNCTION

$-g : \mathbb{R}^d \to \mathbb{R}$ STRONGLY SMOOTH CONVEX FUNCTION

$$\nabla^2 f(x) \preceq MI$$

for all $x \in S$. This upper bound on the Hessian implies for any $x,\ y \in S$,

$$f(y) \leq f(x) + \nabla f(x)^T (y - x) + \frac{M}{2}\|y - x\|_2^2, \tag{9.13}$$

which is analogous to (9.8). Minimizing each side over y yields

$$p^\star \leq f(x) - \frac{1}{2M}\|\nabla f(x)\|_2^2,$$

$$\Updownarrow$$

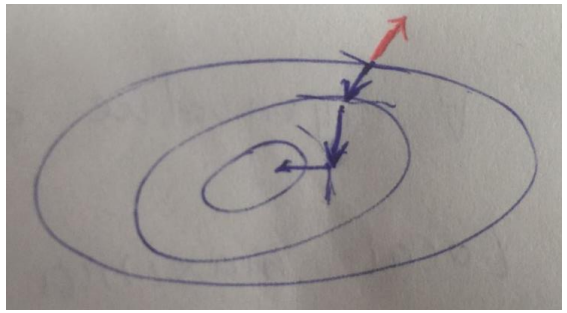
$$p^\star \geq f(x) - \frac{1}{2m}\|\nabla f(x)\|_2^2.$$

$\|f(x) - f^*(x)\|_2 \geq \frac{1}{2M}\|\nabla f(x)\|_2^2$

$\Rightarrow$ If $\nabla f(x)$ is large, $f(x)$ is far from $f^*(x)$

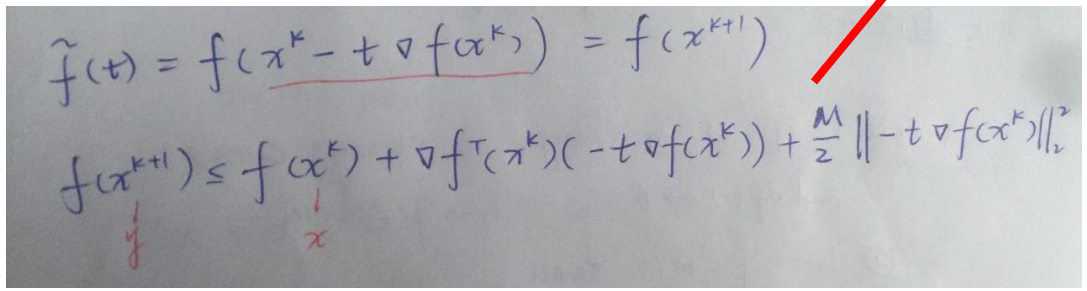
$\|f(x) - f(x^*)\|_2 \leq \frac{1}{2m}\|\nabla f(x)\|_2^2$

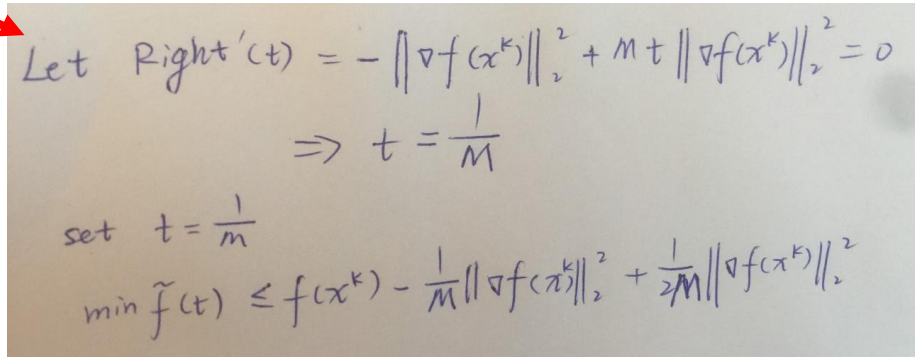
• Gradient Descent – Convergence Guarantees



In this section we present a simple convergence analysis for the gradient method, using the lighter notation $x^+ = x + t\Delta x$ for $x^{(k+1)} = x^{(k)} + t^{(k)}\Delta x^{(k)}$, where $\Delta x = -\nabla f(x)$. We assume f is strongly convex on S, so there are positive constants m and M such that $mI \preceq \nabla^2 f(x) \preceq MI$ for all $x \in S$. Define the function $\tilde{f} : \mathbf{R} \to \mathbf{R}$ by $\tilde{f}(t) = f(x - t\nabla f(x))$, *i.e.*, f as a function of the step length t in the negative gradient direction. In the following discussion we will only consider t for which $x - t\nabla f(x) \in S$. From the inequality (9.13), with $y = x - t\nabla f(x)$, we obtain a quadratic upper bound on $\tilde{f}$:

$$\tilde{f}(t) \le f(x) - t\|\nabla f(x)\|_2^2 + \frac{Mt^2}{2}\|\nabla f(x)\|_2^2. \qquad (9.17)$$





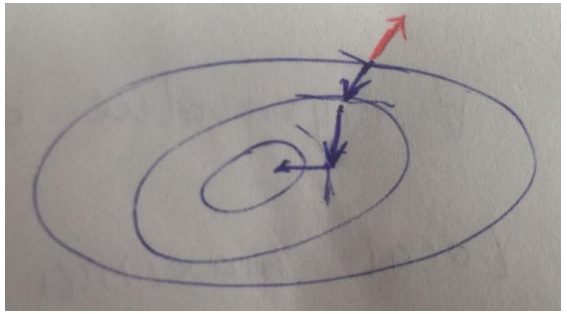
step length that minimizes $\tilde{f}$. The righthand side is a simple quadratic, which is minimized by $t = 1/M$, and has minimum value $f(x) - (1/(2M))\|\nabla f(x)\|_2^2$. Therefore we have

$$f(x^+) = \tilde{f}(t_{\text{exact}}) \le f(x) - \frac{1}{2M}\|\nabla(f(x))\|_2^2.$$

Subtracting $p^\star$ from both sides, we get

$$f(x^+) - p^\star \le f(x) - p^\star - \frac{1}{2M}\|\nabla f(x)\|_2^2.$$

- # Gradient Descent – Convergence Guarantees



Subtracting $p^\star$ from both sides, we get

$$f(x^+) - p^\star \leq f(x) - p^\star - \frac{1}{2M}\|\nabla f(x)\|_2^2.$$

$$\begin{cases} \frac{1}{2M}\|\nabla f(x^k)\|_2^2 + f(x^{k+1}) - p^* \leq f(x^k) - p^* & \times M \\ -\frac{1}{2m}\|\nabla f(x^k)\|_2^2 + f(x^k) - p^* \leq 0 & \times m \end{cases} \quad \oplus$$

$$\Rightarrow M(f(x^{k+1}) - p^*) + m(f(x^k) - p^*) \leq M(f(x^k) - p^*)$$

$$\|f(x) - f(x^*)\|_2 \leq \frac{1}{2m}\|\nabla f(x)\|_2^2$$

We combine this with $\|\nabla f(x)\|_2^2 \geq 2m(f(x) - p^\star)$ (which follows from (9.9)) to conclude

$$f(x^+) - p^\star \leq (1 - m/M)(f(x) - p^\star).$$

$$\left\| \frac{f(x^{k+1}) - p^*}{f(x^k) - p^*} \right\| \leq \left\| 1 - \frac{m}{M} \right\|$$

• Projected Gradient Descent

used in the convex case do not apply directly to non-convex problems. Consider the following optimization problem:

$$\min_{\mathbf{x}\in\mathbb{R}^p} f(\mathbf{x}) \quad \text{s.t. } \mathbf{x}\in\mathcal{C}. \tag{CVX-OPT}$$

In the above optimization problem, $\mathcal{C}\subset\mathbb{R}^p$ is a convex constraint set and $f:\mathbb{R}^p\to\mathbb{R}$ is a convex objective function. We will assume that we have oracle access to the gradient and projection operators, i.e., for any point $\mathbf{x}\in\mathbb{R}^p$ we are able to access $\nabla f(\mathbf{x})$ and $\Pi_{\mathcal{C}}(\mathbf{x})$.

Algorithm 1 Projected Gradient Descent (PGD)

Input: Convex objective f, convex constraint set $\mathcal{C}$, step lengths η_t
Output: A point $\widehat{\mathbf{x}}\in\mathcal{C}$ with near-optimal objective value

1: $\mathbf{x}^1 \leftarrow \mathbf{0}$
2: **for** $t = 1, 2, \ldots, T$ **do**
3: $\quad \mathbf{z}^{t+1} \leftarrow \mathbf{x}^t - \eta_t \cdot \nabla f(\mathbf{x}^t)$
4: $\quad \mathbf{x}^{t+1} \leftarrow \Pi_{\mathcal{C}}(\mathbf{z}^{t+1})$
5: **end for**
6: (OPTION 1) **return** $\widehat{\mathbf{x}}_{\text{final}} = \mathbf{x}^T$
7: (OPTION 2) **return** $\widehat{\mathbf{x}}_{\text{avg}} = (\sum_{t=1}^T \mathbf{x}^t)/T$
8: (OPTION 3) **return** $\widehat{\mathbf{x}}_{\text{best}} = \arg\min_{t\in[T]} f(\mathbf{x}^t)$

• Projected Gradient Descent – Convergence Guarantees

Theorem 2.5. *Let f be a convex objective with bounded gradients and Algorithm 1 be executed for T time steps with step lengths $\eta_t = \eta = \frac{1}{\sqrt{T}}$. Then, for any $\epsilon > 0$, if $T = \mathcal{O}\left(\frac{1}{\epsilon^2}\right)$, then $\frac{1}{T}\sum_{t=1}^{T} f(\mathbf{x}^t) \leq f^* + \epsilon$.*

ensures the function value of the iterates approaches f∗ on an average

Algorithm 1 Projected Gradient Descent (PGD)

Input: Convex objective f, convex constraint set $\mathcal{C}$, step lengths η_t
Output: A point $\hat{\mathbf{x}} \in \mathcal{C}$ with near-optimal objective value

1: $\mathbf{x}^1 \leftarrow \mathbf{0}$
2: **for** $t = 1, 2, \ldots, T$ **do**
3: $\quad \mathbf{z}^{t+1} \leftarrow \mathbf{x}^t - \eta_t \cdot \nabla f(\mathbf{x}^t)$
4: $\quad \mathbf{x}^{t+1} \leftarrow \Pi_{\mathcal{C}}(\mathbf{z}^{t+1})$
5: **end for**
6: (OPTION 1) **return** $\hat{\mathbf{x}}_{\text{final}} = \mathbf{x}^T$
7: (OPTION 2) **return** $\hat{\mathbf{x}}_{\text{avg}} = (\sum_{t=1}^{T} \mathbf{x}^t)/T$
8: (OPTION 3) **return** $\hat{\mathbf{x}}_{\text{best}} = \arg\min_{t \in [T]} f(\mathbf{x}^t)$

algorithm. If we use OPTION 3, i.e., $\hat{\mathbf{x}}_{\text{best}}$, then since by construction, we have $f(\hat{\mathbf{x}}_{\text{best}}) \leq f(\mathbf{x}^t)$ for all t, by applying Theorem 2.5, we get

$$f(\hat{\mathbf{x}}_{\text{best}}) \leq \frac{1}{T}\sum_{t=1}^{T} f(\mathbf{x}^t) \leq f^* + \epsilon,$$

If we use OPTION 2, i.e., $\hat{\mathbf{x}}_{\text{avg}}$, which is cheaper since we do not have to perform function evaluations to find the best iterate, we can apply Jensen's inequality (Lemma 2.1) to get the following

$$f(\hat{\mathbf{x}}_{\text{avg}}) = f\left(\frac{1}{T}\sum_{t=1}^{T} \mathbf{x}^t\right) \leq \frac{1}{T}\sum_{t=1}^{T} f(\mathbf{x}^t) \leq f^* + \epsilon.$$

Lemma 2.1 (Jensen's Inequality). *If X is a random variable taking values in the domain of a convex function f, then $\mathbb{E}[f(X)] \geq f(\mathbb{E}[X])$*

• Projected Gradient Descent – Convergence Guarantees

(Apply Convexity) We apply convexity to upper bound the potential function at every step. Convexity is a global property and very useful in getting an upper bound on the level of sub-optimality of the current iterate in such analyses.

$$\Phi_t = f(\mathbf{x}^t) - f(\mathbf{x}^*) \leq \left\langle \nabla f(\mathbf{x}^t), \mathbf{x}^t - \mathbf{x}^* \right\rangle$$

$$f(y) \geq f(x) + \nabla f(x)^T (y - x) + \frac{m}{2}\|y - x\|_2^2$$

We now do some elementary manipulations

$$\begin{aligned}
\left\langle \nabla f(\mathbf{x}^t), \mathbf{x}^t - \mathbf{x}^* \right\rangle &= \frac{1}{\eta} \left\langle \eta \cdot \nabla f(\mathbf{x}^t), \mathbf{x}^t - \mathbf{x}^* \right\rangle \\
&= \frac{1}{2\eta} \left(\left\| \mathbf{x}^t - \mathbf{x}^* \right\|_2^2 + \left\| \eta \cdot \nabla f(\mathbf{x}^t) \right\|_2^2 - \left\| \mathbf{x}^t - \eta \cdot \nabla f(\mathbf{x}^t) - \mathbf{x}^* \right\|_2^2 \right) \\
&= \frac{1}{2\eta} \left(\left\| \mathbf{x}^t - \mathbf{x}^* \right\|_2^2 + \left\| \eta \cdot \nabla f(\mathbf{x}^t) \right\|_2^2 - \left\| \mathbf{z}^{t+1} - \mathbf{x}^* \right\|_2^2 \right) \\
&\leq \frac{1}{2\eta} \left(\left\| \mathbf{x}^t - \mathbf{x}^* \right\|_2^2 + \eta^2 G^2 - \left\| \mathbf{z}^{t+1} - \mathbf{x}^* \right\|_2^2 \right),
\end{aligned}$$

a b a-b

$2ab = a^2 + b^2 - (a+b)^2$

$\mathbf{z}^{t+1} \leftarrow \mathbf{x}^t - \eta_t \cdot \nabla f(\mathbf{x}^t)$

$\|\nabla f(\mathbf{x})\|_2 \leq G$

where the first step applies the identity $2ab = a^2 + b^2 - (a+b)^2$, the second step uses the update step of the PGD algorithm that sets $\mathbf{z}^{t+1} \leftarrow \mathbf{x}^t - \eta_t \cdot \nabla f(\mathbf{x}^t)$, and the third step uses the fact that the objective function f has bounded gradients.

• Projected Gradient Descent – Convergence Guarantees

(Apply Projection Property) We apply Lemma 2.4 to get

$$\left\|\mathbf{z}^{t+1}-\mathbf{x}^*\right\|_2^2 \geq \left\|\mathbf{x}^{t+1}-\mathbf{x}^*\right\|_2^2$$

Lemma 2.4 (Projection Property-II). *For any convex set $\mathcal{C} \subset \mathbb{R}^p$ and any $\mathbf{z} \in \mathbb{R}^p$, let $\widehat{\mathbf{z}} := \Pi_{\mathcal{C}}(\mathbf{z})$. Then for all $\mathbf{x} \in \mathcal{C}$, $\|\widehat{\mathbf{z}}-\mathbf{x}\|_2 \leq \|\mathbf{z}-\mathbf{x}\|_2$.*

Putting all these together gives us

$$f(\mathbf{x}^t)-f(\mathbf{x}^*) = \Phi_t \leq \frac{1}{2\eta}\left(\left\|\mathbf{x}^t-\mathbf{x}^*\right\|_2^2-\left\|\mathbf{x}^{t+1}-\mathbf{x}^*\right\|_2^2\right)+\frac{\eta G^2}{2}$$

$$\Phi_t = f(\mathbf{x}^t)-f(\mathbf{x}^*) \leq \left\langle \nabla f(\mathbf{x}^t), \mathbf{x}^t-\mathbf{x}^*\right\rangle \leq \frac{1}{2\eta}\left(\left\|\mathbf{x}^t-\mathbf{x}^*\right\|_2^2+\eta^2G^2-\left\|\mathbf{z}^{t+1}-\mathbf{x}^*\right\|_2^2\right)$$

Analysis

The above expression is interesting since it tells us that, apart from the $\eta G^2/2$ term which is small as $\eta = \frac{1}{\sqrt{T}}$, the current sub-optimality Φ_t is small if the consecutive iterates $\mathbf{x}^t$ and $\mathbf{x}^{t+1}$ are close to each other (and hence similar in distance from $\mathbf{x}^*$).

once PGD stops making a lot of progress, it actually converges to the optimum! Since we are using a constant step length, only a vanishing gradient can cause PGD to stop progressing. For convex functions, this only happens at global optima.

$$\mathbf{z}^{t+1} \leftarrow \mathbf{x}^t - \eta_t \cdot \nabla f(\mathbf{x}^t)$$
$$\mathbf{x}^{t+1} \leftarrow \Pi_{\mathcal{C}}(\mathbf{z}^{t+1})$$

for convex functions, this only happens at global optima. Summing the expression up across time steps, performing telescopic cancellations, using $\mathbf{x}^1 = \mathbf{0}$, and dividing throughout by T gives us

$$f(\mathbf{x}^t)-f(\mathbf{x}^*)$$

$$\frac{1}{T}\sum_{t=1}^{T}\Phi_t \leq \frac{1}{2\eta T}\left(\|\mathbf{x}^*\|_2^2-\|\mathbf{x}^{T+1}-\mathbf{x}^*\|_2^2\right)+\frac{\eta G^2}{2} \qquad \|\mathbf{x}^{t+1}-\mathbf{x}^*\|_2 \geq 0$$

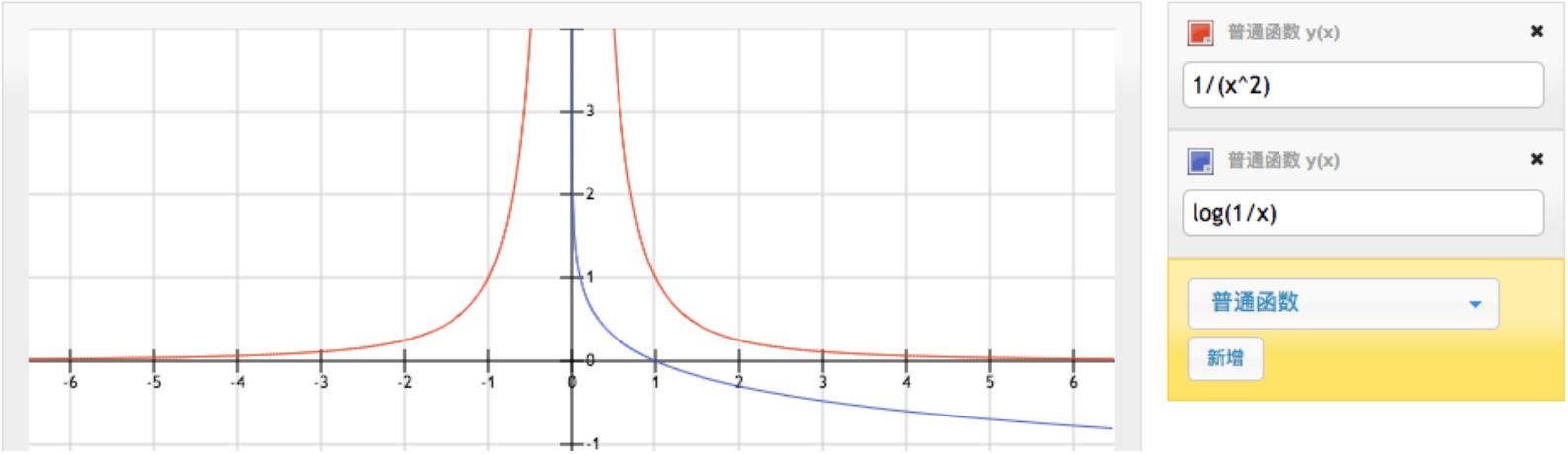
$$\leq \frac{1}{2\sqrt{T}}\left(\|\mathbf{x}^*\|_2^2+G^2\right), \qquad \eta = 1/\sqrt{T}.$$

$$\frac{1}{T}\sum_{t=1}^{T} f(\mathbf{x}^t) \leq f^* + \epsilon$$

• Projected Gradient Descent – Convergence Guarantees

Theorem 2.6. *Let f be an objective that satisfies the α-SC and β-SS properties. Let Algorithm 1 be executed with step lengths $\eta_t = \eta = \frac{1}{\beta}$. Then after at most $T = \mathcal{O}\left(\frac{\beta}{\alpha}\log\frac{\beta}{\epsilon}\right)$ steps, we have $f(\mathbf{x}^T) \leq f(\mathbf{x}^*) + \epsilon$.*

This result is particularly nice since it ensures that the final iterate $\widehat{\mathbf{x}}_{\text{final}} = \mathbf{x}^T$ converges, allowing us to use OPTION 1 in Algorithm 1 when the objective is SC/SS. A further advantage is the accelerated rate of convergence. Whereas for general convex functions, PGD requires $\mathcal{O}\left(\frac{1}{\epsilon^2}\right)$ iterations to reach an ϵ-optimal solution, for SC/SS functions, it requires only $\mathcal{O}\left(\log\frac{1}{\epsilon}\right)$ iterations.



• Projected Gradient Descent – Convergence Guarantees

$$f(\mathbf{x}^T) \leq f(\mathbf{x}^*) + \left\langle \nabla f(\mathbf{x}^*), \mathbf{x}^T - \mathbf{x}^* \right\rangle + \frac{\beta}{2} \left\| \mathbf{x}^T - \mathbf{x}^* \right\|_2^2.$$

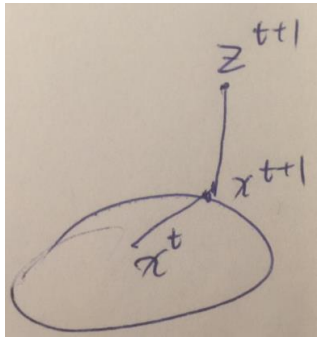
Now, since $\mathbf{x}^*$ is an optimal point for the constrained optimization problem with a convex constraint set $\mathcal{C}$, the first order optimality condition [see Bubeck, 2015, Proposition 1.3] gives us $\langle \nabla f(\mathbf{x}^*), \mathbf{x} - \mathbf{x}^* \rangle \leq 0$ for any $\mathbf{x} \in \mathcal{C}$. Applying this condition with $\mathbf{x} = \mathbf{x}^T$ gives us

$$f(\mathbf{x}^T) - f(\mathbf{x}^*) \leq \frac{\beta}{2} \left\| \mathbf{x}^T - \mathbf{x}^* \right\|_2^2 \leq \epsilon,$$

which proves that $\mathbf{x}^T$ is an ϵ-optimal point. We now show $\left\| \mathbf{x}^T - \mathbf{x}^* \right\|_2^2 \leq \frac{2\epsilon}{\beta}$.

(Apply Strong Smoothness) As discussed before, we use it to show that PGD always makes significant progress in each iteration.

$$\begin{aligned} f(\mathbf{x}^{t+1}) - f(\mathbf{x}^t) &\leq \left\langle \nabla f(\mathbf{x}^t), \mathbf{x}^{t+1} - \mathbf{x}^t \right\rangle + \frac{\beta}{2} \left\| \mathbf{x}^t - \mathbf{x}^{t+1} \right\|_2^2 \\ &= \left\langle \nabla f(\mathbf{x}^t), \mathbf{x}^{t+1} - \mathbf{x}^* \right\rangle + \left\langle \nabla f(\mathbf{x}^t), \mathbf{x}^* - \mathbf{x}^t \right\rangle + \frac{\beta}{2} \left\| \mathbf{x}^t - \mathbf{x}^{t+1} \right\|_2^2 \\ &= \frac{1}{\eta} \left\langle \mathbf{x}^t - \mathbf{z}^{t+1}, \mathbf{x}^{t+1} - \mathbf{x}^* \right\rangle + \left\langle \nabla f(\mathbf{x}^t), \mathbf{x}^* - \mathbf{x}^t \right\rangle + \frac{\beta}{2} \left\| \mathbf{x}^t - \mathbf{x}^{t+1} \right\|_2^2 \end{aligned}$$



(Apply Projection Rule) The above expression contains an unwieldy term $\mathbf{z}^{t+1}$. Since this term only appears during projection steps, we eliminate it by applying Projection Property-I (Lemma 2.3) to get

$$\begin{aligned} \left\langle \mathbf{x}^t - \mathbf{z}^{t+1}, \mathbf{x}^{t+1} - \mathbf{x}^* \right\rangle &\leq \left\langle \mathbf{x}^t - \mathbf{x}^{t+1}, \mathbf{x}^{t+1} - \mathbf{x}^* \right\rangle \\ &= \frac{\left\| \mathbf{x}^t - \mathbf{x}^* \right\|_2^2 - \left\| \mathbf{x}^t - \mathbf{x}^{t+1} \right\|_2^2 - \left\| \mathbf{x}^{t+1} - \mathbf{x}^* \right\|_2^2}{2} \end{aligned}$$

Then for all $\mathbf{x} \in \mathcal{C}$, $\langle \mathbf{x} - \widehat{\mathbf{z}}, \mathbf{z} - \widehat{\mathbf{z}} \rangle \leq 0$.

?

• Projected Gradient Descent – Convergence Guarantees

(1)
$$f(\mathbf{x}^{t+1}) - f(\mathbf{x}^t) \leq \left\langle \nabla f(\mathbf{x}^t), \mathbf{x}^{t+1} - \mathbf{x}^t \right\rangle + \frac{\beta}{2} \left\| \mathbf{x}^t - \mathbf{x}^{t+1} \right\|_2^2$$
$$= \left\langle \nabla f(\mathbf{x}^t), \mathbf{x}^{t+1} - \mathbf{x}^* \right\rangle + \left\langle \nabla f(\mathbf{x}^t), \mathbf{x}^* - \mathbf{x}^t \right\rangle + \frac{\beta}{2} \left\| \mathbf{x}^t - \mathbf{x}^{t+1} \right\|_2^2$$
$$= \frac{1}{\eta} \left\langle \mathbf{x}^t - \mathbf{z}^{t+1}, \mathbf{x}^{t+1} - \mathbf{x}^* \right\rangle + \left\langle \nabla f(\mathbf{x}^t), \mathbf{x}^* - \mathbf{x}^t \right\rangle + \frac{\beta}{2} \left\| \mathbf{x}^t - \mathbf{x}^{t+1} \right\|_2^2$$

(2)
$$\left\langle \mathbf{x}^t - \mathbf{z}^{t+1}, \mathbf{x}^{t+1} - \mathbf{x}^* \right\rangle \leq \left\langle \mathbf{x}^t - \mathbf{x}^{t+1}, \mathbf{x}^{t+1} - \mathbf{x}^* \right\rangle$$
$$= \frac{\left\| \mathbf{x}^t - \mathbf{x}^* \right\|_2^2 - \left\| \mathbf{x}^t - \mathbf{x}^{t+1} \right\|_2^2 - \left\| \mathbf{x}^{t+1} - \mathbf{x}^* \right\|_2^2}{2}$$

Using $\eta = 1/\beta$ and combining the above results gives us

$$f(\mathbf{x}^{t+1}) - f(\mathbf{x}^t) \leq \left\langle \nabla f(\mathbf{x}^t), \mathbf{x}^* - \mathbf{x}^t \right\rangle + \frac{\beta}{2} \left(\left\| \mathbf{x}^t - \mathbf{x}^* \right\|_2^2 - \left\| \mathbf{x}^{t+1} - \mathbf{x}^* \right\|_2^2 \right)$$

(Apply Strong Convexity) The above expression is perfect for a telescoping step but for the inner product term. Fortunately, this can be eliminated using strong convexity.

$$\left\langle \nabla f(\mathbf{x}^t), \mathbf{x}^* - \mathbf{x}^t \right\rangle \leq f(\mathbf{x}^*) - f(\mathbf{x}^t) - \frac{\alpha}{2} \left\| \mathbf{x}^t - \mathbf{x}^* \right\|_2^2$$

Combining with the above this gives us

$$f(\mathbf{x}^{t+1}) - f(\mathbf{x}^*) \leq \frac{\beta - \alpha}{2} \left\| \mathbf{x}^t - \mathbf{x}^* \right\|_2^2 - \frac{\beta}{2} \left\| \mathbf{x}^{t+1} - \mathbf{x}^* \right\|_2^2.$$

• Projected Gradient Descent – Convergence Guarantees

Combining with the above this gives us

$$f(\mathbf{x}^{t+1}) - f(\mathbf{x}^*) \leq \frac{\beta - \alpha}{2} \left\|\mathbf{x}^t - \mathbf{x}^*\right\|_2^2 - \frac{\beta}{2} \left\|\mathbf{x}^{t+1} - \mathbf{x}^*\right\|_2^2.$$

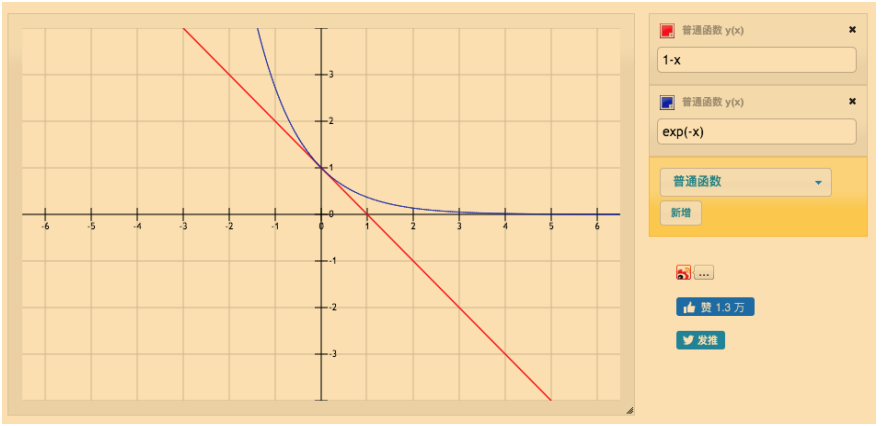
$f(\mathbf{x}^{t+1}) \geq f(\mathbf{x}^*)$. This means

$$\frac{\beta}{2} \left\|\mathbf{x}^{t+1} - \mathbf{x}^*\right\|_2^2 \leq \frac{\beta - \alpha}{2} \left\|\mathbf{x}^t - \mathbf{x}^*\right\|_2^2,$$

which can be written as

$$\Phi_{t+1} \leq \left(1 - \frac{\alpha}{\beta}\right) \Phi_t \leq \exp\left(-\frac{\alpha}{\beta}\right) \Phi_t,$$

$$\Phi_t = \left\|\mathbf{x}^t - \mathbf{x}^*\right\|_2^2.$$



where we have used the fact that $1 - x \leq \exp(-x)$ for all $x \in \mathbb{R}$. What we have arrived at is a very powerful result as it assures us that the potential value goes down by a constant fraction at every iteration! Applying this result recursively gives us

$$\Phi_{t+1} \leq \exp\left(-\frac{\alpha t}{\beta}\right) \Phi_1 = \exp\left(-\frac{\alpha t}{\beta}\right) \|\mathbf{x}^*\|_2^2,$$

since $\mathbf{x}^1 = \mathbf{0}$. Thus, we deduce that $\Phi_T = \left\|\mathbf{x}^T - \mathbf{x}^*\right\|_2^2 \leq \frac{2\epsilon}{\beta}$ after at most $T = \mathcal{O}\left(\frac{\beta}{\alpha} \log \frac{\beta}{\epsilon}\right)$ steps which finishes the proof □

• Projected Gradient Descent – Convergence Guarantees

We notice that the convergence of the PGD algorithm is of the form $\|\mathbf{x}^{t+1} - \mathbf{x}^*\|_2^2 \leq \exp\left(-\frac{\alpha t}{\beta}\right)\|\mathbf{x}^*\|_2^2$. The number $\kappa := \frac{\beta}{\alpha}$ is the *condition number* of the optimization problem.

Definition 2.6 (Condition Number - Informal). *The condition number of a function $f : \mathcal{X} \to \mathbb{R}$ is a scalar $\kappa \in \mathbb{R}$ that bounds how much the function value can change relative to a perturbation of the input.*

Functions with a small condition number are stable and changes to their input do not affect the function output values too much. However, functions with a large condition number can be quite jumpy and experience abrupt changes in output values even if the input is changed slightly. To gain a deeper appreciation of this concept, consider a differentiable function f that is also α-SC and β-SS. Consider a stationary point for f i.e., a point $\mathbf{x}$ such that $\nabla f(\mathbf{x}) = \mathbf{0}$. For a general function, such a point can be a local optima or a saddle point. However, since f is strongly convex, $\mathbf{x}$ is the (unique) global minima[5] of f. Then we have, for any other point $\mathbf{y}$

$$\frac{\alpha}{2}\|\mathbf{x} - \mathbf{y}\|_2^2 \leq f(\mathbf{y}) - f(\mathbf{x}) \leq \frac{\beta}{2}\|\mathbf{x} - \mathbf{y}\|_2^2$$

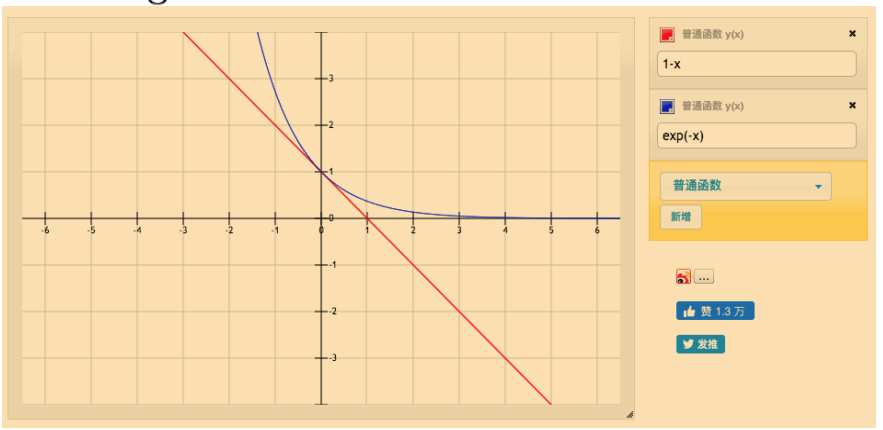
Dividing throughout by $\frac{\alpha}{2}\|\mathbf{x} - \mathbf{y}\|_2^2$ gives us

$$\frac{f(\mathbf{y}) - f(\mathbf{x})}{\frac{\alpha}{2}\|\mathbf{x} - \mathbf{y}\|_2^2} \in \left[1, \frac{\beta}{\alpha}\right] := [1, \kappa]$$

• Projected Gradient Descent – Convergence Guarantees

$$\frac{f(\mathbf{y}) - f(\mathbf{x})}{\frac{\alpha}{2}\|\mathbf{x}-\mathbf{y}\|_2^2} \in \left[1, \frac{\beta}{\alpha}\right] := [1, \kappa]$$

Thus, upon perturbing the input from the global minimum $\mathbf{x}$ to a point $\|\mathbf{x}-\mathbf{y}\|_2 =: \epsilon$ distance away, the function value does change much – it goes up by an amount at least $\frac{\alpha\epsilon^2}{2}$ but at most $\kappa \cdot \frac{\alpha\epsilon^2}{2}$. Such well behaved response to perturbations is very easy for optimization algorithms to exploit to give fast convergence.



$$\Phi_{t+1} \leq \left(1 - \frac{\alpha}{\beta}\right)\Phi_t \leq \exp\left(-\frac{\alpha}{\beta}\right)\Phi_t,$$

The condition number of the objective function can significantly affect the convergence rate of algorithms. Indeed, if $\kappa = \frac{\beta}{\alpha}$ is small, then $\exp\left(-\frac{\alpha}{\beta}\right) = \exp\left(-\frac{1}{\kappa}\right)$ would be small, ensuring fast convergence. However, if $\kappa \gg 1$ then $\exp\left(-\frac{1}{\kappa}\right) \approx 1$ and the procedure might offer slow convergence.

- **Chapter 3 Non-Convex PGD**

Part II

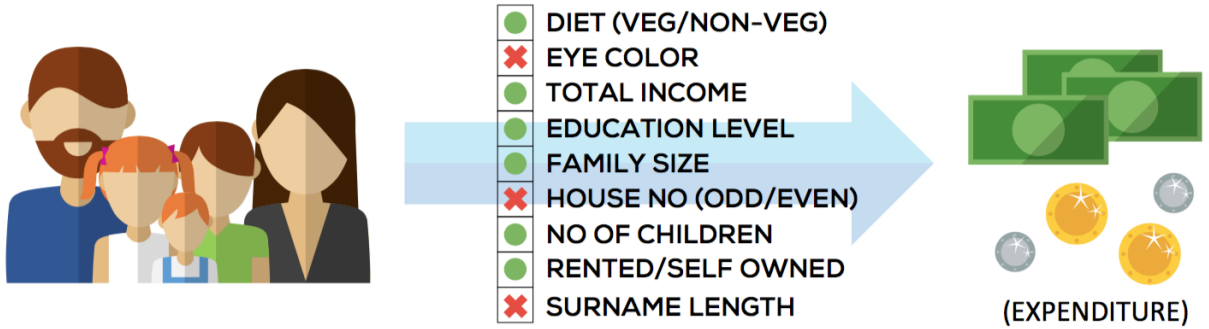
Non-convex Optimization Primitives

Executing the projected gradient descent algorithm with non-convex problems requires projections onto non-convex sets. Now, a quick look at the projection problem

$$\Pi_{\mathcal{C}}(\mathbf{z}) := \arg\min_{\mathbf{x}\in\mathcal{C}} \|\mathbf{x}-\mathbf{z}\|_2$$

reveals that this is an optimization problem in itself. Thus, when the set $\mathcal{C}$ to be projected onto is non-convex, the projection problem can itself be NP-hard. However, for several well-structured sets, projection can be carried out efficiently despite the sets being non-convex.

• 3.1.1 Projecting into Sparse Vectors



In the sparse linear regression example discussed in § 1,

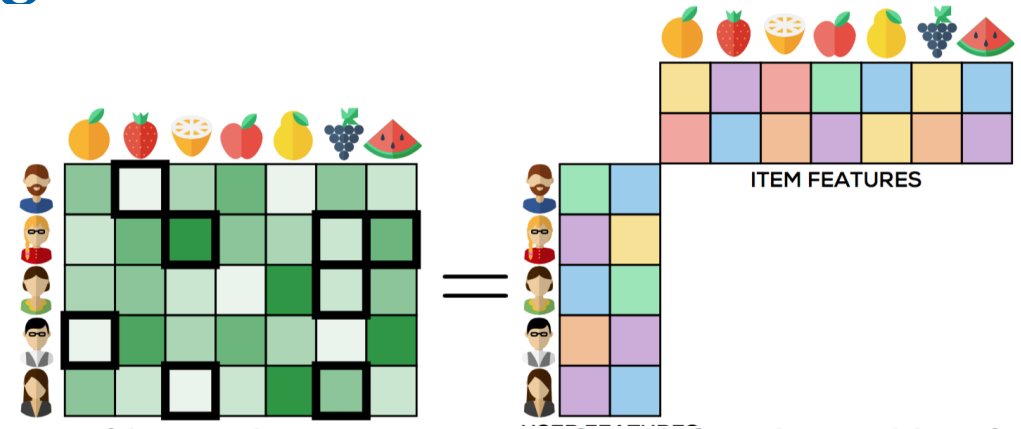
$$\widehat{\mathbf{w}} = \underset{\|\mathbf{w}\|_0 \le s}{\arg\min} \sum_{i=1}^{n} \left(y_i - \mathbf{x}_i^\top \mathbf{w}\right)^2,$$

applying projected gradient descent requires projections onto the set of s-sparse vectors i.e., $\mathcal{B}_0(s) := \{\mathbf{x} \in \mathbb{R}^p, \|\mathbf{x}\|_0 \le s\}$. The following result shows that the projection $\Pi_{\mathcal{B}_0(s)}(\mathbf{z})$ can be carried out by simply sorting the coordinates of the vector $\mathbf{z}$ according to magnitude and setting all except the top-s coordinates to zero.

Lemma 3.1. *For any vector $\mathbf{z} \in \mathbb{R}^p$, let σ be the permutation that sorts the coordinates of $\mathbf{z}$ in decreasing order of magnitude, i.e., $\left|\mathbf{z}_{\sigma(1)}\right| \ge \left|\mathbf{z}_{\sigma(2)}\right| \ge \ldots \ge \left|\mathbf{z}_{\sigma(p)}\right|$. Then the vector $\widehat{\mathbf{z}} := \Pi_{\mathcal{B}_0(s)}(\mathbf{z})$ is obtained by setting $\widehat{\mathbf{z}}_i = \mathbf{z}_i$ if $\sigma(i) \le s$ and $\widehat{\mathbf{z}}_i = 0$ otherwise.*

Proof. We first notice that since the function $x \mapsto x^2$ is an increasing function on the positive half of the real line, we have $\underset{\mathbf{x}\in\mathcal{C}}{\arg\min} \|\mathbf{x} - \mathbf{z}\|_2 = \underset{\mathbf{x}\in\mathcal{C}}{\arg\min} \|\mathbf{x} - \mathbf{z}\|_2^2$. Next, we observe that the vector $\widehat{\mathbf{z}} := \Pi_{\mathcal{B}_0(s)}$ must satisfy $\widehat{\mathbf{z}}_i = \mathbf{z}_i$ for all $i \in \mathrm{supp}(\widehat{\mathbf{z}})$ otherwise we can decrease the objective value $\|\widehat{\mathbf{z}} - \mathbf{z}\|_2^2$ by ensuring this. Having established this gives us $\|\widehat{\mathbf{z}} - \mathbf{z}\|_2^2 = \sum_{i \notin \mathrm{supp}(\widehat{\mathbf{z}})} \mathbf{z}_i^2$. This is clearly minimized when $\mathrm{supp}(\widehat{\mathbf{z}})$ has the coordinates of $\mathbf{z}$ with largest magnitude. □

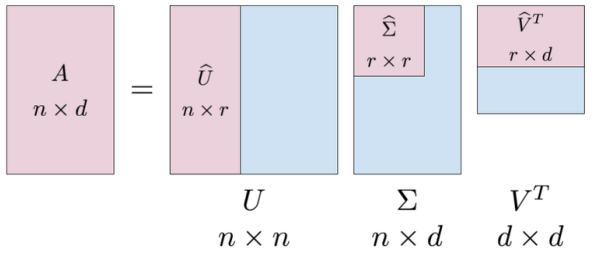
3.1.2 Projecting into Low-rank Matrices



we need to project onto the set of low-rank matrices. Let us first define this problem formally. Consider matrices of a certain order, say $m \times n$ and let $\mathcal{C} \subset \mathbb{R}^{m \times n}$ be an arbitrary set of matrices. Then, the projection operator $\Pi_{\mathcal{C}}(\cdot)$ is defined as follows: for any matrix $A \in R^{m \times n}$,

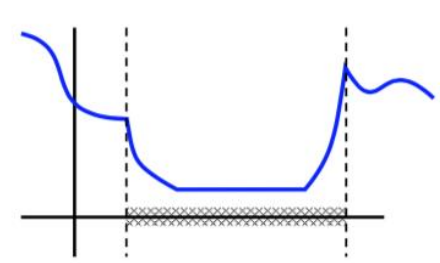
$$\Pi_{\mathcal{C}}(A) := \arg\min_{X \in \mathcal{C}} \|A - X\|_F ,$$

where $\|\cdot\|_F$ is the Frobenius norm over matrices. For low rank projections we require $\mathcal{C}$ to be the set of low rank matrices $\mathcal{B}_{\text{rank}}(r) := \{A \in \mathbb{R}^{m \times n}, \text{rank}(A) \leq r\}$. Yet again, this projection can be done efficiently by performing a *Singular Value Decomposition* on the matrix A and retaining the top r singular values and vectors. The Eckart-Young-Mirsky theorem proves that this indeed gives us the projection.



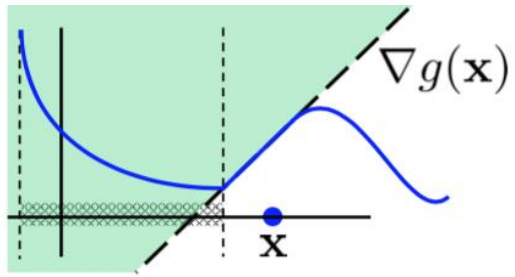
Theorem 3.2 (Eckart-Young-Mirsky theorem). *For any matrix $A \in \mathbb{R}^{m \times n}$, let $U\Sigma V^\top$ be the singular value decomposition of A such that $\Sigma = diag(\sigma_1, \sigma_2, \ldots, \sigma_{\min(m,n)})$ where $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_{\min(m,n)}$. Then for any $r \leq \min(m, n)$, the matrix $\widehat{A}_{(r)} := \Pi_{\mathcal{B}_{rank}(r)}(A)$ can be obtained as $U_{(r)}\Sigma_{(r)}V_{(r)}^\top$ where $U_{(r)} := [U_1 U_2 \ldots U_r]$, $V(r) := [V_1 V_2 \ldots V_r]$, and $\Sigma_{(r)} := diag(\sigma_1, \sigma_2, \ldots, \sigma_r)$.*

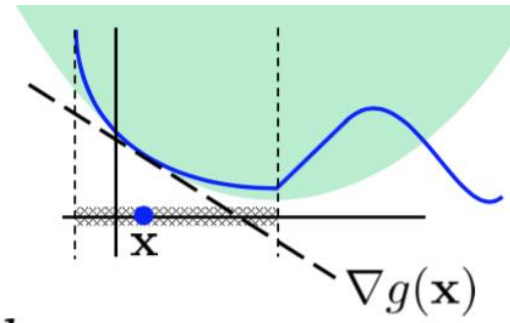
- ## 3.2 Restricted Strong Convexity and Smoothness



$-f : \mathbb{R}^d \to \mathbb{R}$

RESTRICTED CONVEXITY





$-g : \mathbb{R}^d \to \mathbb{R}$

A NON-CONVEX FUNCTION THAT SATISFIES RESTRICTED STRONG CONVEXITY

Definition 3.1 (Restricted Convexity). *A continuously differentiable function $f : \mathbb{R}^p \to \mathbb{R}$ is said to satisfy restricted convexity over a (possibly non-convex) region $\mathcal{C} \subseteq \mathbb{R}^p$ if for every $\mathbf{x}, \mathbf{y} \in \mathcal{C}$ we have $f(\mathbf{y}) \geq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle$, where $\nabla f(\mathbf{x})$ is the gradient of f at $\mathbf{x}$.*

Definition 3.2 (Restricted Strong Convexity/Smoothness). *A continuously differentiable function $f : \mathbb{R}^p \to \mathbb{R}$ is said to satisfy α-restricted strong convexity (RSC) and β-restricted strong smoothness (RSS) over a (possibly non-convex) region $\mathcal{C} \subseteq \mathbb{R}^p$ if for every $\mathbf{x}, \mathbf{y} \in \mathcal{C}$, we have*

$$\frac{\alpha}{2} \|\mathbf{x} - \mathbf{y}\|_2^2 \leq f(\mathbf{y}) - f(\mathbf{x}) - \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle \leq \frac{\beta}{2} \|\mathbf{x} - \mathbf{y}\|_2^2 .$$

3.3 Generalized PGD

Algorithm 2 Generalized Projected Gradient Descent (gPGD)

Input: Objective function f, constraint set $\mathcal{C}$, step length η
Output: A point $\widehat{\mathbf{x}} \in \mathcal{C}$ with near-optimal objective value

1: $\mathbf{x}^1 \leftarrow \mathbf{0}$
2: **for** $t = 1, 2, \ldots, T$ **do**
3: $\quad \mathbf{z}^{t+1} \leftarrow \mathbf{x}^t - \eta \cdot \nabla f(\mathbf{x}^t)$
4: $\quad \mathbf{x}^{t+1} \leftarrow \Pi_{\mathcal{C}}(\mathbf{z}^{t+1})$
5: **end for**
6: **return** $\widehat{\mathbf{x}}_{\text{final}} = \mathbf{x}^T$

Assume that $\nabla f(\mathbf{x}_*) = \mathbf{0}$. This assumption is satisfied whenever the objective function is differentiable and the optimal point $\mathbf{x}$ lies in the interior of the constraint set C

Theorem 3.3. *Let f be a (possibly non-convex) function satisfying the α-RSC and β-RSS properties over a (possibly non-convex) constraint set $\mathcal{C}$ with $\beta/\alpha < 2$. Let Algorithm 2 be executed with a step length $\eta = \frac{1}{\beta}$. Then after at most $T = \mathcal{O}\left(\frac{\alpha}{2\alpha-\beta} \log \frac{1}{\epsilon}\right)$ steps, $f(\mathbf{x}^T) \leq f(\mathbf{x}^*) + \epsilon$.*

This result holds even when the step length is set to values that are large enough but yet smaller than $1/\beta$. However, setting $\eta = \frac{1}{\beta}$ simplifies the proof and allows us to focus on the key concepts.

- ## 3.3 Generalized PGD

Proof (of Theorem 3.3). Recall that the proof of Theorem 2.5 used the SC/SS properties for the analysis. We will replace these by the RSC/RSS properties – we will use RSC to track the global convergence of the algorithm and RSS to locally assess the progress made by the algorithm in each iteration. We will use $\Phi_t = f(\mathbf{x}^{t+1}) - f(\mathbf{x}^*)$ as the potential function.

(Apply Restricted Strong Smoothness) Since both $\mathbf{x}^t, \mathbf{x}^{t+1} \in \mathcal{C}$ due to the projection steps, we apply the β-RSS property to them.

$$\begin{aligned} f(\mathbf{x}^{t+1}) - f(\mathbf{x}^t) &\leq \left\langle \nabla f(\mathbf{x}^t), \mathbf{x}^{t+1} - \mathbf{x}^t \right\rangle + \frac{\beta}{2} \left\| \mathbf{x}^t - \mathbf{x}^{t+1} \right\|_2^2 \\ &= \frac{1}{\eta} \left\langle \mathbf{x}^t - \mathbf{z}^{t+1}, \mathbf{x}^{t+1} - \mathbf{x}^t \right\rangle + \frac{\beta}{2} \left\| \mathbf{x}^t - \mathbf{x}^{t+1} \right\|_2^2 \\ &= \frac{\beta}{2} \left(\left\| \mathbf{x}^{t+1} - \mathbf{z}^{t+1} \right\|_2^2 - \left\| \mathbf{x}^t - \mathbf{z}^{t+1} \right\|_2^2 \right) \end{aligned}$$

Notice that this step crucially uses the fact that $\eta = 1/\beta$.

3.3 Generalized PGD

(Apply Projection Property) We are again stuck with the unwieldy $\mathbf{z}^{t+1}$ term. However, unlike before, we cannot apply projection properties I or II as non-convex projections do not satisfy them. Instead, we resort to Projection Property-O (Lemma 2.2), that all projections (even non-convex ones) must satisfy. Applying this property gives us

(Projection Property-O)
$\|\widehat{\mathbf{z}} - \mathbf{z}\|_2 \leq \|\mathbf{x} - \mathbf{z}\|_2$

(Projection Property-II)
$\|\widehat{\mathbf{z}} - \mathbf{x}\|_2 \leq \|\mathbf{z} - \mathbf{x}\|_2$

$$
\begin{aligned}
f(\mathbf{x}^{t+1}) - f(\mathbf{x}^t) &\leq \frac{\beta}{2}\left(\left\|\mathbf{x}^* - \mathbf{z}^{t+1}\right\|_2^2 - \left\|\mathbf{x}^t - \mathbf{z}^{t+1}\right\|_2^2\right) \\
&= \frac{\beta}{2}\left(\left\|\mathbf{x}^* - \mathbf{x}^t\right\|_2^2 + 2\left\langle \mathbf{x}^* - \mathbf{x}^t, \mathbf{x}^t - \mathbf{z}^{t+1}\right\rangle\right) \\
&= \frac{\beta}{2}\left\|\mathbf{x}^* - \mathbf{x}^t\right\|_2^2 + \left\langle \mathbf{x}^* - \mathbf{x}^t, \nabla f(\mathbf{x}^t)\right\rangle
\end{aligned}
$$

Xt+1 is the project of zt+1

$\eta = \frac{1}{\beta}$

(Apply Restricted Strong Convexity) Since both $\mathbf{x}^t, \mathbf{x}^* \in \mathcal{C}$, we apply the α-RSC property to them. However, we do so in two ways:

$$
\begin{aligned}
f(\mathbf{x}^*) - f(\mathbf{x}^t) &\geq \left\langle \nabla f(\mathbf{x}^t), \mathbf{x}^* - \mathbf{x}^t\right\rangle + \frac{\alpha}{2}\left\|\mathbf{x}^t - \mathbf{x}^*\right\|_2^2 \\
f(\mathbf{x}^t) - f(\mathbf{x}^*) &\geq \left\langle \nabla f(\mathbf{x}^*), \mathbf{x}^t - \mathbf{x}^*\right\rangle + \frac{\alpha}{2}\left\|\mathbf{x}^t - \mathbf{x}^*\right\|_2^2 \geq \frac{\alpha}{2}\left\|\mathbf{x}^t - \mathbf{x}^*\right\|_2^2,
\end{aligned}
$$

where in the second line we used the fact that we assumed $\nabla f(\mathbf{x}^*) = \mathbf{0}$. We recall that this assumption can be done away with but makes the proof more complicated which we wish to avoid. Simple manipulations with the two equations give us

?

$$
\left\langle \nabla f(\mathbf{x}^t), \mathbf{x}^* - \mathbf{x}^t\right\rangle + \frac{\beta}{2}\left\|\mathbf{x}^* - \mathbf{x}^t\right\|_2^2 \leq \left(2 - \frac{\beta}{\alpha}\right)\left(f(\mathbf{x}^*) - f(\mathbf{x}^t)\right)
$$

- ## 3.3 Generalized PGD

Putting this in the earlier expression gives us

$$f(\mathbf{x}^{t+1}) - f(\mathbf{x}^t) \leq \left(2 - \frac{\beta}{\alpha}\right)\left(f(\mathbf{x}^*) - f(\mathbf{x}^t)\right)$$

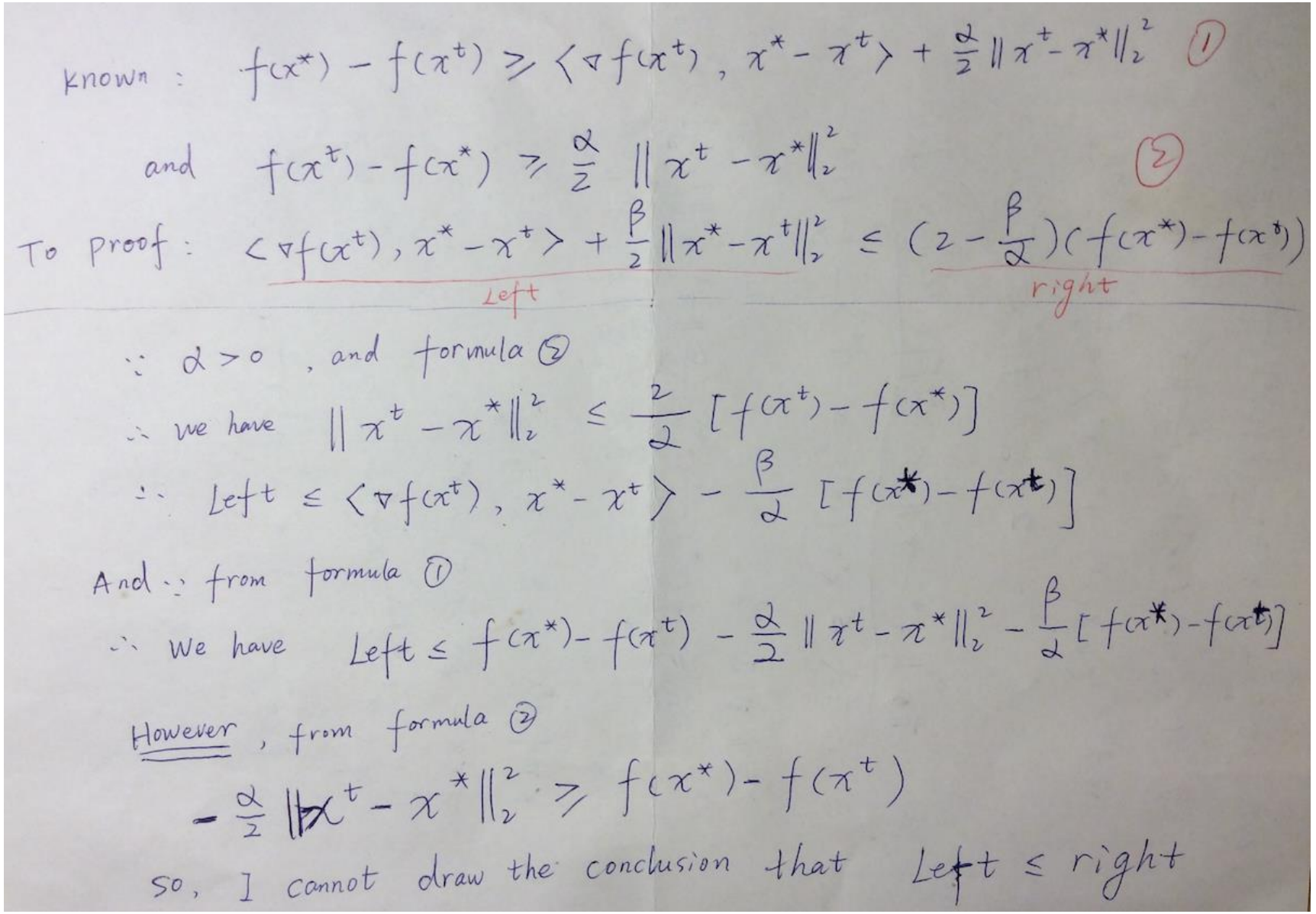
The above inequality is quite interesting. It tells us that the larger the gap between $f(\mathbf{x}^*)$ and $f(\mathbf{x}^t)$, the larger will be the drop in objective value in going from $\mathbf{x}^t$ to $\mathbf{x}^{t+1}$. The form of the result is also quite fortunate as it assures us that we will cover a constant fraction $\left(2 - \frac{\beta}{\alpha}\right)$ of the remaining "distance" to $\mathbf{x}^*$ at each step! Rearranging this gives

$$\Phi_{t+1} \leq (\kappa - 1)\Phi_t, \qquad \Phi_t = f(\mathbf{x}^{t+1}) - f(\mathbf{x}^*)$$

where $\kappa = \beta/\alpha$. Note that we always have $\kappa \geq 1$[4] and by assumption $\kappa = \beta/\alpha < 2$, so that we always have $\kappa - 1 \in [0, 1)$. This proves the result after simple manipulations. □

require $\kappa < 2$.

- **3.3 Generalized PGD**



known : $f(x^*) - f(x^t) \geq \langle \nabla f(x^t), x^* - x^t \rangle + \frac{\alpha}{2}\|x^t - x^*\|_2^2$ ①

and $f(x^t) - f(x^*) \geq \frac{\alpha}{2}\|x^t - x^*\|_2^2$ ②

To proof : $\underbrace{\langle \nabla f(x^t), x^* - x^t \rangle + \frac{\beta}{2}\|x^* - x^t\|_2^2}_{\text{Left}} \leq \underbrace{(2 - \frac{\beta}{\alpha})(f(x^*) - f(x^t))}_{\text{right}}$

$\because \alpha > 0$, and formula ②

$\therefore$ we have $\|x^t - x^*\|_2^2 \leq \frac{2}{\alpha}[f(x^t) - f(x^*)]$

$\therefore$ Left $\leq \langle \nabla f(x^t), x^* - x^t \rangle - \frac{\beta}{\alpha}[f(x^*) - f(x^t)]$

And $\because$ from formula ①

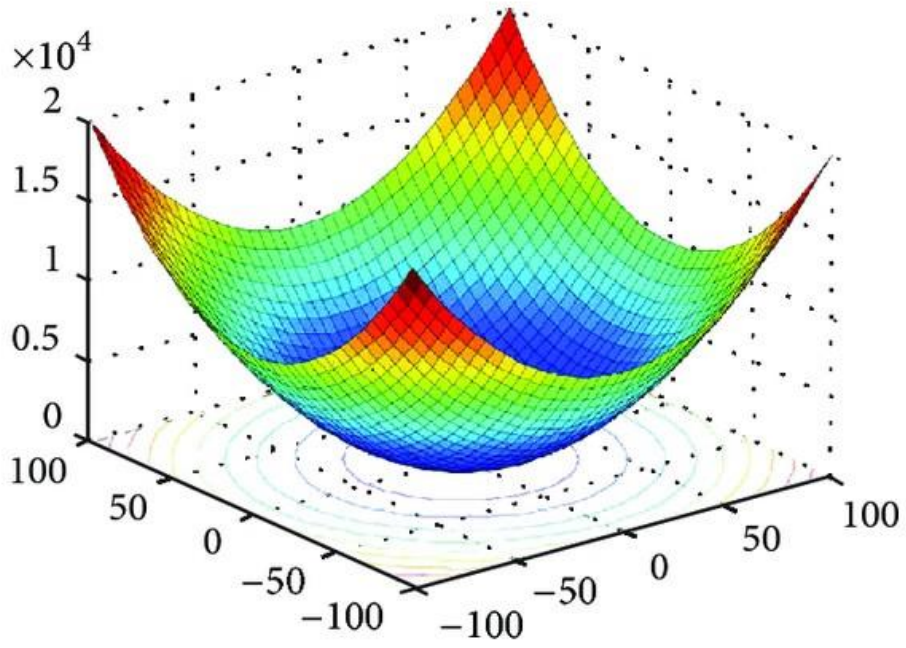
$\therefore$ we have Left $\leq f(x^*) - f(x^t) - \frac{\alpha}{2}\|x^t - x^*\|_2^2 - \frac{\beta}{\alpha}[f(x^*) - f(x^t)]$

However , from formula ②

$-\frac{\alpha}{2}\|x^t - x^*\|_2^2 \geq f(x^*) - f(x^t)$

so, I cannot draw the conclusion that Left $\leq$ right

- # Chapter 4 Alternating Minimasation



Definition 4.1 (Joint Convexity). *A continuously differentiable function in two variables* $f : \mathbb{R}^p \times \mathbb{R}^q \to \mathbb{R}$ *is considered jointly convex if for every* $(\mathbf{x}^1, \mathbf{y}^1), (\mathbf{x}^2, \mathbf{y}^2) \in \mathbb{R}^p \times \mathbb{R}^q$ *we have*

$$f(\mathbf{x}^2, \mathbf{y}^2) \geq f(\mathbf{x}^1, \mathbf{y}^1) + \left\langle \nabla f(\mathbf{x}^1, \mathbf{y}^1), (\mathbf{x}^2, \mathbf{y}^2) - (\mathbf{x}^1, \mathbf{y}^1) \right\rangle,$$

where $\nabla f(\mathbf{x}^1, \mathbf{y}^1)$ *is the gradient of* f *at the point* $(\mathbf{x}^1, \mathbf{y}^1)$.

- # Chapter 4 Alternating Minimasation

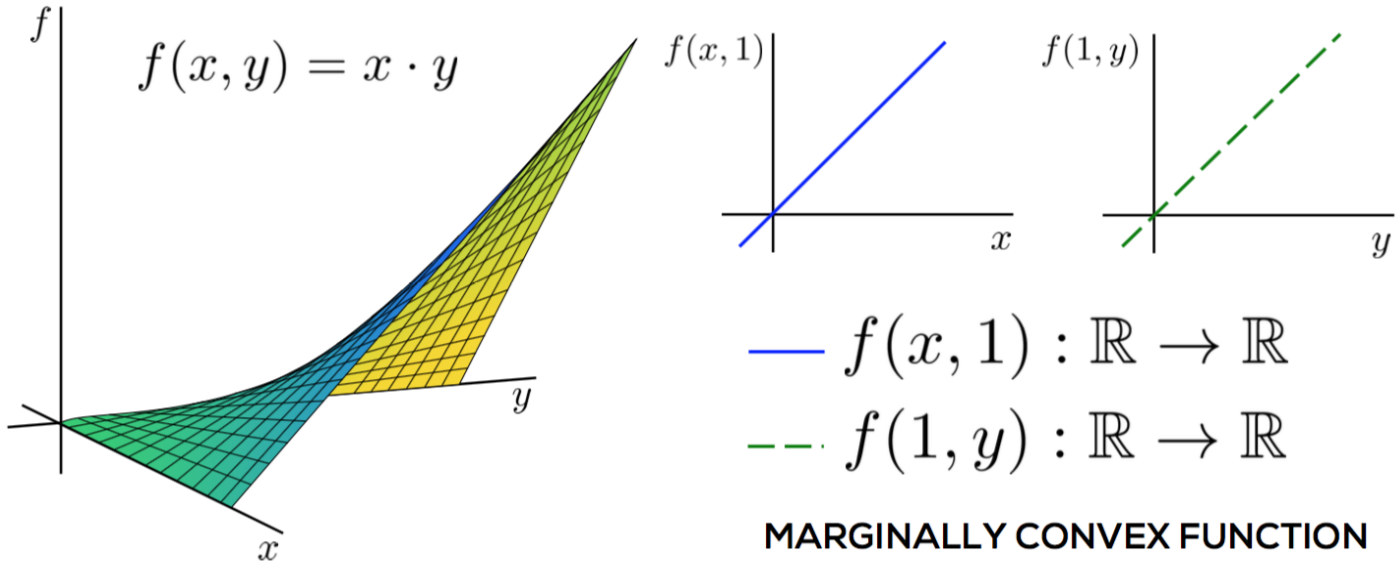


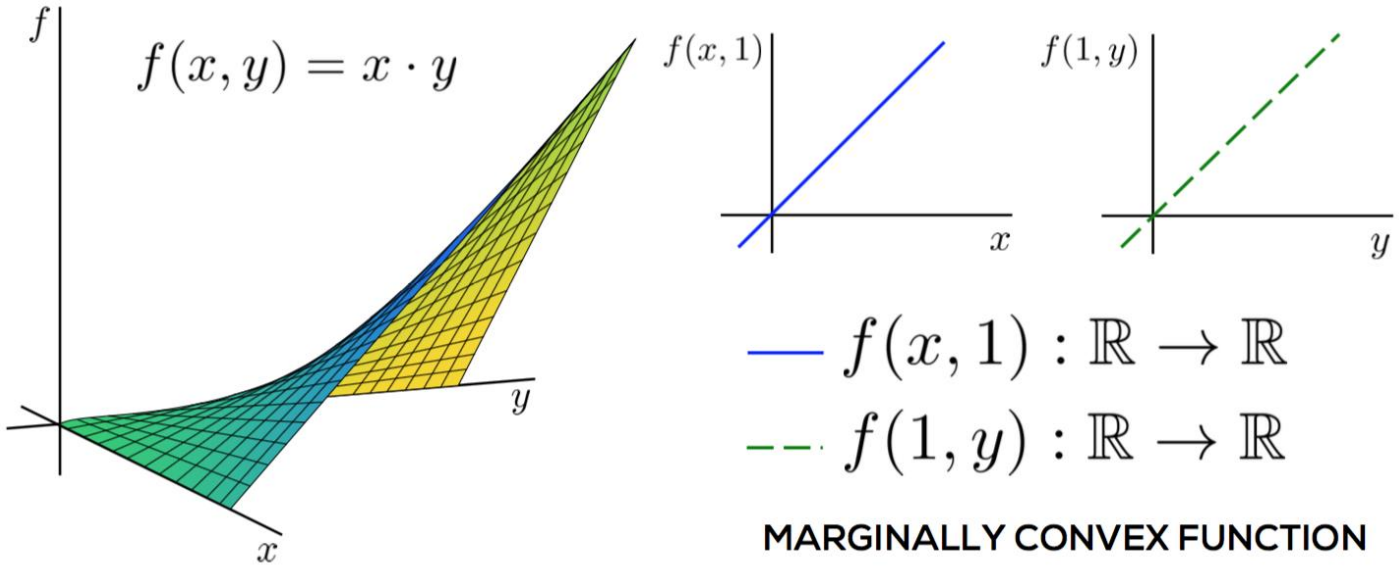
Figure 4.1: A marginally convex function is not necessarily (jointly) convex. The function $f(x,y) = x \cdot y$ is marginally linear, hence marginally convex, in both its variables, but clearly not a (jointly) convex function.

Definition 4.2 (Marginal Convexity). *A continuously differentiable function of two variables $f : \mathbb{R}^p \times \mathbb{R}^q \to \mathbb{R}$ is considered marginally convex in its first variable if for every value of $\mathbf{y} \in \mathbb{R}^q$, the function $f(\cdot, \mathbf{y}) : \mathbb{R}^p \to \mathbb{R}$ is convex, i.e., for every $\mathbf{x}^1, \mathbf{x}^2 \in \mathbb{R}^p$, we have*

$$f(\mathbf{x}^2, \mathbf{y}) \geq f(\mathbf{x}^1, \mathbf{y}) + \left\langle \nabla_{\mathbf{x}} f(\mathbf{x}^1, \mathbf{y}), \mathbf{x}^2 - \mathbf{x}^1 \right\rangle,$$

where $\nabla_{\mathbf{x}} f(\mathbf{x}^1, \mathbf{y})$ is the partial gradient of f with respect to its first variable at the point $(\mathbf{x}^1, \mathbf{y})$. A similar condition is imposed for f to be considered marginally convex in its second variable.

- # Chapter 4 Alternating Minimasation



Definition 4.3 (Marginally Strongly Convex/Smooth Function). *A continuously differentiable function $f : \mathbb{R}^p \times \mathbb{R}^q \to \mathbb{R}$ is considered (uniformly) α-marginally strongly convex (MSC) and (uniformly) β-marginally strongly smooth (MSS) in its first variable if for every value of $\mathbf{y} \in \mathbb{R}^q$, the function $f(\cdot, \mathbf{y}) : \mathbb{R}^p \to \mathbb{R}$ is α-strongly convex and β-strongly smooth, i.e., for every $\mathbf{x}^1, \mathbf{x}^2 \in \mathbb{R}^p$, we have*

$$\frac{\alpha}{2}\left\|\mathbf{x}^2 - \mathbf{x}^1\right\|_2^2 \leq f(\mathbf{x}^2, \mathbf{y}) - f(\mathbf{x}^1, \mathbf{y}) - \left\langle \mathbf{g}, \mathbf{x}^2 - \mathbf{x}^1 \right\rangle \leq \frac{\beta}{2}\left\|\mathbf{x}^2 - \mathbf{x}^1\right\|_2^2,$$

where $\mathbf{g} = \nabla_{\mathbf{x}} f(\mathbf{x}^1, \mathbf{y})$ is the partial gradient of f with respect to its first variable at the point $(\mathbf{x}^1, \mathbf{y})$. A similar condition is imposed for f to be considered (uniformly) MSC/MSS in its second variable.

- **Chapter 4 Alternating Minimasation**

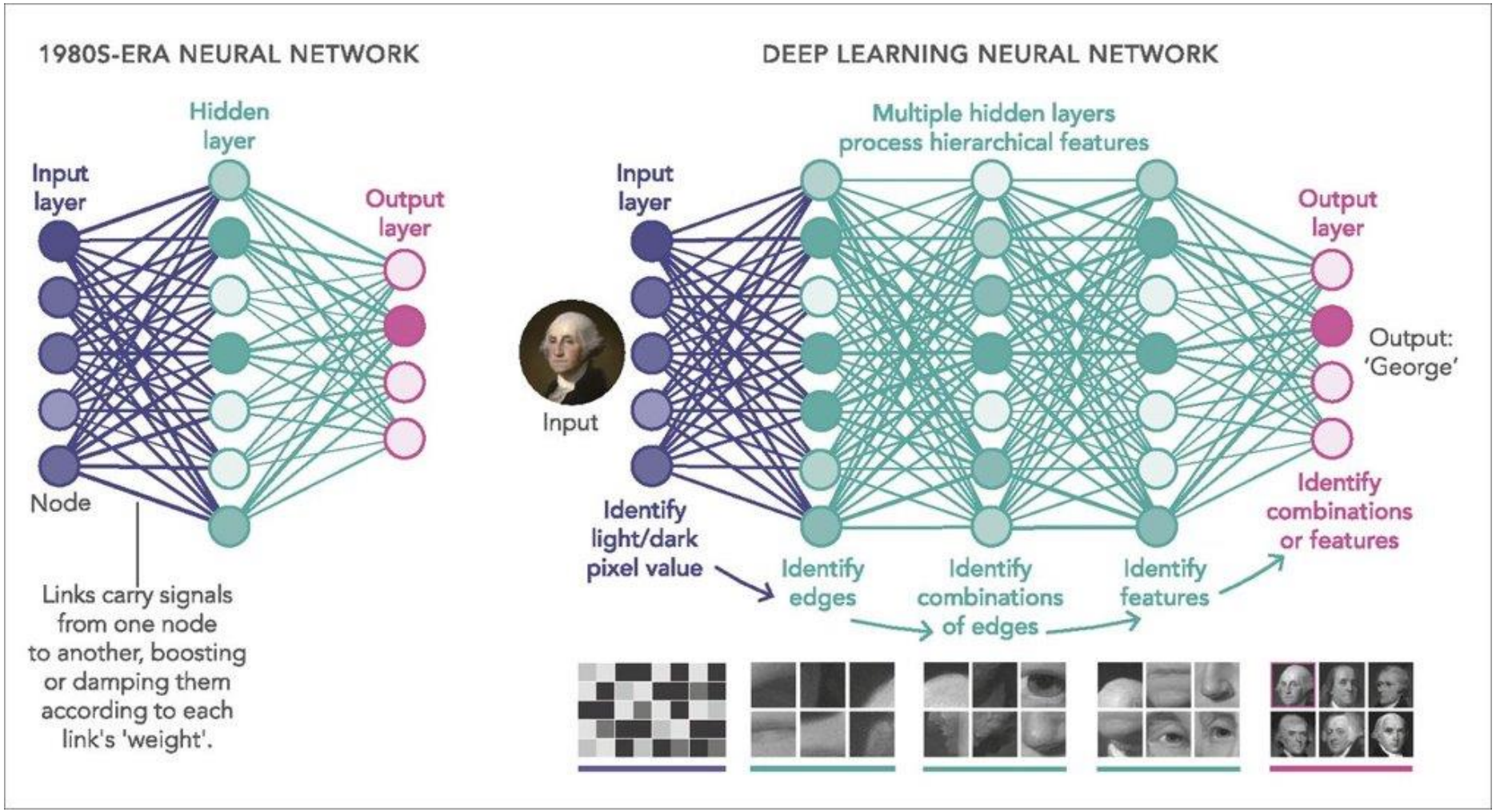
Algorithm 3 Generalized Alternating Minimization (gAM)

Input: Objective function $f : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$

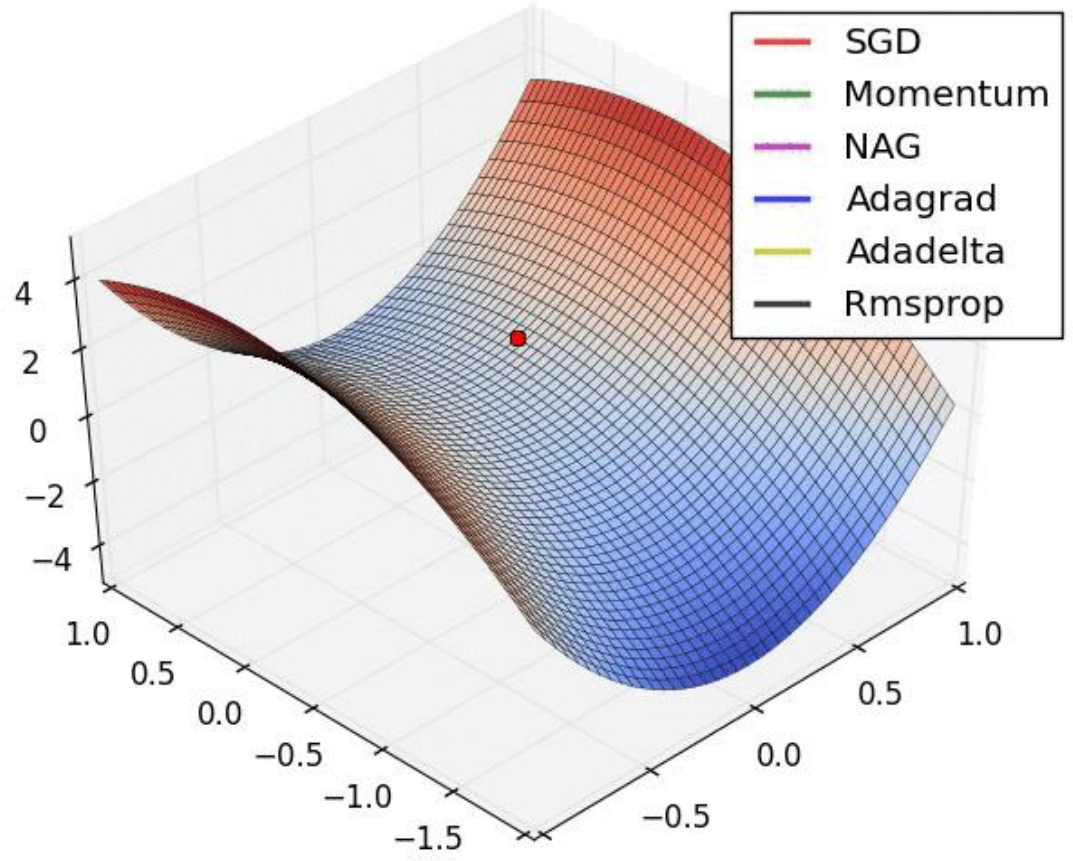
Output: A point $(\widehat{\mathbf{x}}, \widehat{\mathbf{y}}) \in \mathcal{X} \times \mathcal{Y}$ with near-optimal objective value

1: $(\mathbf{x}^1, \mathbf{y}^1) \leftarrow$ INITALIZE()
2: **for** $t = 1, 2, \ldots, T$ **do**
3: $\quad \mathbf{x}^{t+1} \leftarrow \arg\min_{\mathbf{x} \in \mathcal{X}} f(\mathbf{x}, \mathbf{y}^t)$
4: $\quad \mathbf{y}^{t+1} \leftarrow \arg\min_{\mathbf{y} \in \mathcal{Y}} f(\mathbf{x}^{t+1}, \mathbf{y})$
5: **end for**
6: **return** $(\mathbf{x}^T, \mathbf{y}^T)$

- **Optimising NN is a non-convex problem**



- **Optimising NN is a non-convex problem**



Thank you!