AM 221: Advanced Optimization Spring 2018

Dr. Rasmus Kyng *Lecture 1 — Monday, January 22th*

1 Overview

In this course we will cover optimization through the perspective of convex optimization, along with important applications. We will also see some combinatorial optimization, including submodular optimization, which is a discrete analogue of convex optimization.

Convex optimization is a central tool for solving large-scale problems which in recent years has had a profound impact on statistical machine learning, data analysis, mathematical finance, signal processing, control, approximation algorithms, operations research, as well as many other areas.

In a single sentence, the premise of this course can be summarized as follows:

Optimization is an elegant mathematical theory which provides fundamental tools for reasoning about and solving problems across a broad range of areas in the data sciences.

The first part of the course will be dedicated the theory of convex optimization and applications of this theory in other areas. We will then take a detour into combinatorial optimization, before returning to some advanced topics in convex optimization. The course is theoretical, but both exercises and the final project may involve some programming.

2 Examples from Statistics and Machine Learning

Say we want to predict people's height, based on some other information about them, e.g. the height of their mother. We will use a set of observations of heights of a person and their mother, and use this to build a model for the relationship between these variables. Using this model, we can try to predict the heights of new people based on the height of their mother.

Consider a set observations $(x_1, y_1), \ldots, (x_m, y_m)$, where $y_i \in \mathbb{R}$ denotes the height of person i and $x_i \in \mathbb{R}$ denotes the height of their mother. We will use a simple model for the relation between x_i and y_i, namely a linear relationship, i.e. $y_i = ax_i + b$, for some parameters $a, b \in \mathbb{R}$. This model is called *linear regression*. We will estimate the parameters by finding values a^*, b^* that minimize the residual sum of squares in our data set:

$$(a^*, b^*) = \text{argmin}_{a,b\in\mathbb{R}} \sum_{i=1}^{m} (y_i - (a \cdot x_i - b))^2$$

Figure 1 shows as an example a set of observations, along with the line $y = ax + b$, and the residuals $y_i - (ax_i + b)$.

Figure 2 shows a real data set on the relation

Suppose for simplicity that $b^* = 0$. Then, a^*will be the value that set the derivative $\frac{\partial}{\partial a} \sum_{i=1}^{m} (y_i - a \cdot x_i)^2$ to zero. Thus

$$\frac{\partial}{\partial a} \sum_{i=1}^{m} (y_i - a \cdot x_i)^2 = 2 \sum_{i=1}^{m} a x_i^2 - y_i x_i,$$

gives $a^* = \frac{\sum_{i=1}^{m} y_i x_i}{\sum_{i=1}^{m} x_i^2}$.

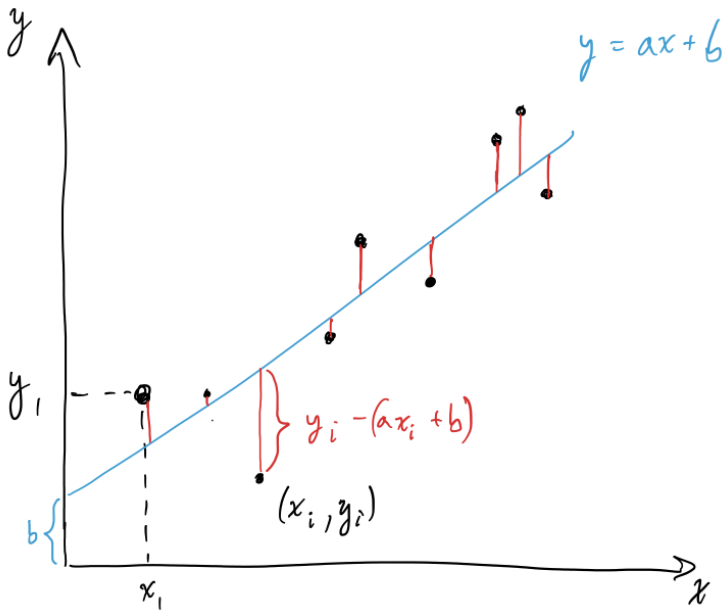

Figure 1: Linear regression and residual.

Linear regression in multiple dimensions. It's easy to imagine that having more data about each individual in the above example could let us better predict their height. For example, it might help to know the height of the person's father, their biological sex, and age. More generally, we could encode any number of features of the person as real numbers, giving a d-dimensional point $\mathbf{x} \in \mathbb{R}^d$ for each person. The linear regression model assumes the output $y \in \mathbb{R}$ is linear in $\mathbf{x}$, i.e. $y = \mathbf{a}^\top \mathbf{x} + b$, for some $b \in \mathbb{R}$ and $\mathbf{a} \in \mathbb{R}^d$.

It is convenient to include b in the vector of coefficients by considering each data point $\mathbf{x}$ as a vector in $\mathbb{R}^{d+1}$ with the first coordinate being 1 for every point and the remaining being our original $\mathbf{x}$. Now with $\mathbf{a} \in \mathbb{R}^{d+1}$, we can write $\mathbf{y} = \mathbf{a}^\top \mathbf{x}$, and the first coordinate of $\mathbf{a}$ will give us the value of the offset b.

For a data set $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)$, the residual sum of squares of for a parameter vector $\mathbf{a}$ is then

$$\mathrm{RSS}(\mathbf{a}) = \sum_{i=1}^{m} (y_i - \mathbf{a}^\top \mathbf{x}_i)^2$$

and our goal is to find a vector of parameters $\mathbf{a}^* \in \operatorname{argmin}_{\mathbf{a} \in \mathbb{R}^d} \mathrm{RSS}(\mathbf{a})$. In statistics, one of often studies how to best fit parameters to describe or predict data based on models of relationships

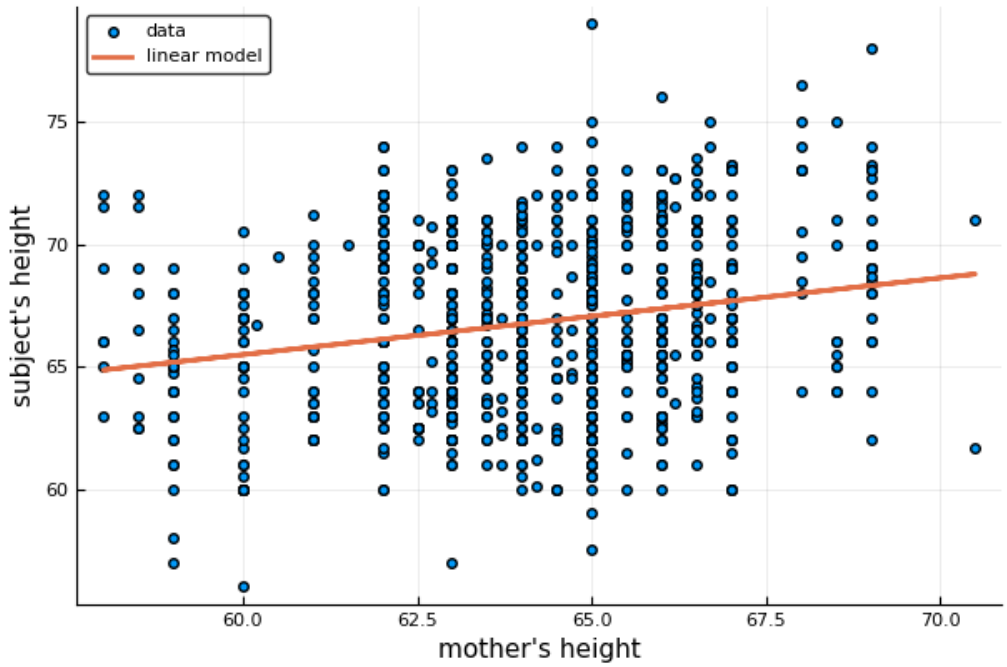

Figure 2: Fitting a linear model of the relationship between the height of the subject's mother, and the subject's own height. The data used here was first used by the famous statistician Francis Galton in 1886, when he analyzed it using linear regression. The units are inches.

between inputs $\mathbf{x}$ and output y. Usually these models include noise, for example, linear regression with Gaussian noise would say that for each data point i, we have $y_i = \mathbf{a}^\top \mathbf{x}_i + \xi_i$, where ξ_i is a random normally distributed variable, independent of the noise at other datapoints.

Statistics is often concerned with formulating sensible optimization criteria that when solved lead to good parameter estimates. For example, in linear regression with Gaussian noise, depending on ones goals, estimating parameters by minimizing RSS is often a good choice.

In contrast, the main focus of this course is to understand what kinds of optimization problems can be efficiently solved and to understand the algorithms we use to solve these problems.

Sparse linear regression. Suppose we have a linear model $y = \mathbf{a}^\top \mathbf{x}$, and we want to estimate the parameter vector $\mathbf{a}$ from a (noisy) data set $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)$ with $\mathbf{x}_i \in \mathbb{R}^d$.

What should we do if we expect that in fact y only depends on a few of the coordinates of $\mathbf{x}$? This could happen if we have collected a lot more "features" for each of our data points than we expect actually are relevant for predicting y.

In this situation, it might make sense to ask for the best fitting parameter vector $\mathbf{a} \in \mathbb{R}^d$ with at

most k non-zero entries, for $k \ll d$. Formally, the problem we have to solve is then

$$\begin{aligned} \min & \sum_{i=1}^{m} (y_i - \mathbf{a}^\top \mathbf{x}_i)^2 \\ & \text{support}(\mathbf{a}) \subseteq S \\ \text{s.t. } & |S| \leq k. \end{aligned}$$

Since this optimization problem involves a discrete choice of the support of $\mathbf{a}$, it is a *combinatorial optimization problem*, also sometimes referred to as a discrete optimization problem, or an *integer program*.

Unfortunately, many variants of sparse linear regression are NP-hard to solve. The same is true of many discrete optimization problems, but often one can find approximately optimal solutions, or assumptions on the input that make the problem tractable.

LASSO: Convex optimization under constraints. It turns out that when performing linear regression, we can encourage our optimization problem to give us sparse parameter vectors as solutions by requiring the solution has small ℓ_1-norm, i.e. $\|\mathbf{a}\|_1 = \sum_{j=1}^{d} |\mathbf{a}(j)|$ is small.

$$\begin{aligned} \min & \sum_{i=1}^{m} (y_i - \mathbf{a}^\top \mathbf{x}_i)^2 \\ \text{s.t. } & \sum_{j=1}^{d} |\mathbf{a}(j)| \leq t. \end{aligned}$$

LASSO tends to yield sparse solutions, i.e. where some coordinates of $\mathbf{a}$ are 0. This can lead to better prediction accuracy and often gives more easily interpretable results.

In fact, under some conditions of the sample data, it is sometimes possible to prove that LASSO gives the same answer that sparse linear regression would give, i.e. the best fit parameter vector with small support. This is one of the central results in the literature on *compressed sensing*.

LASSO is an example of optimization *under constraints*. We imposed additional requirements on our solution beyond merely having a small value of our target objective function.

3 Optimization Problems

The example of fitting parameters is a special case of an *optimization problem*: minimizing (or maximizing) a function f under some constraints. The function that we seek to minimize or maximize is referred to as the *objective*.

Focusing for now on optimization over $\mathbf{x} \in \mathbb{R}^d$, we usually write optimization problems as:

$$\begin{aligned} \min_{\mathbf{x}\in\mathbb{R}^d} \text{ (or max) } & f(\mathbf{x}) \\ s.t. \ & g_1(\mathbf{x}) \leq b_1 \\ & . \\ & . \\ & . \\ & g_m(\mathbf{x}) \leq b_m \end{aligned}$$

where $\{g_i(\mathbf{x})\}_{i=1}^m$ encode the constraints. In the LASSO case, there was only one constraint function $g(\mathbf{x}) = \sum_{i=1}^d |\mathbf{x}(i)|$. The set of points which respect the constraints is called the *feasible set*.

Definition. *For a given optimization problem the set* $\mathcal{F} = \{\mathbf{x} \in \mathbb{R}^n \ : \ g_i(\mathbf{x}) \leq b_i, \forall i \in [m]\}$ *is called the* ***feasible set****. A point* $\mathbf{x} \in \mathcal{F}$ *is called a* ***feasible point****, and a point* $\mathbf{x}' \notin \mathcal{F}$ *is called an* ***infeasible point****.*

Ideally, would would like to find optimal solutions for the optimization problems we consider. Let's define what we mean exactly.

Definition. *For a* maximization *problem* $\mathbf{x}^\star$ *is called an* ***optimal solution*** *if* $f(\mathbf{x}^\star) \geq f(\mathbf{x})$, $\forall \mathbf{x} \in \mathcal{F}$. *Similarly, for a* minimization *problem* $\mathbf{x}^\star$ *is an optimal solution if* $f(\mathbf{x}^\star) \leq f(\mathbf{x})$, $\forall \mathbf{x} \in \mathcal{F}$.

What happens if there are *no feasible points*? In this case, an optimal solution cannot exist, and we say the problem is infeasible.

Definition. *If* $\mathcal{F} = \emptyset$ *we say that the optimization problem is* ***infeasible****. If* $\mathcal{F} \neq \emptyset$ *we say the optimization problem is* ***feasible****.*

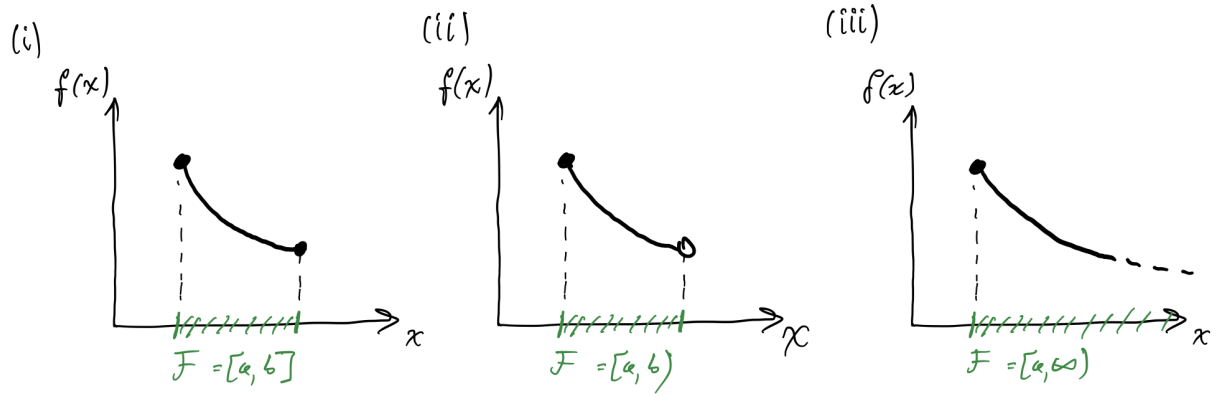

Figure 3

Consider three examples depicted in Figure 3:

(i) $\mathcal{F} = [a, b]$

(ii) $\mathcal{F} = [a, b)$

(iii) $\mathcal{F} = [a, \infty)$

In the first example, the minimum of the function is attained at b. In the second case the region is open and therefore there is no minimum function value, since for every point we will choose, there will always be another point with a smaller function value. Lastly, in the third example, the region in unbounded and the function decreasing, thus again there will always be another point with a smaller function value.

3.1 Sufficient Condition for Optimality

The following theorem, which is a fundamental theorem in real analysis, gives us a sufficient (though not necessary) condition for optimality. Before stating the theorem, let's first recall the Bolzano-Weierstrass theorem from real analysis (which you will prove in section this week).

Theorem. *(Bolzano-Weierstrass) Every bounded sequence in $\mathbb{R}^n$ has a convergent subsequence.*

Secondly, we recall the boundedness theorem:

Theorem. *(Boundedness Theorem) Let $f : \mathbb{R}^n \to \mathbb{R}$ be a continuous function and $\mathcal{F} \subseteq \mathbb{R}^n$ be nonempty, bounded, and closed. Then f is bounded on $\mathcal{F}$.*

Theorem (Extreme Value Theorem)**.** *Let $f : \mathbb{R}^n \to \mathbb{R}$ be a continuous function and $\mathcal{F} \subseteq \mathbb{R}^n$ be nonempty, bounded, and closed. Then, the optimization problem $\min f(\mathbf{x}) : \mathbf{x} \in \mathcal{F}$ has an optimal solution.*

Proof. Let α be the infimum of f over $\mathcal{F}$ (i.e. the largest value for which any point $\mathbf{x} \in \mathcal{F}$ respects $f(\mathbf{x}) \geq \alpha$); by the Boundedness Theorem, such a value exists, as f is lower-bounded, and the set of lower bounds has a greatest lower bound, α.

Let

$$\mathcal{F}_k := \{\mathbf{x} \in \mathcal{F} : \alpha \leq f(\mathbf{x}) \leq \alpha + 2^{-k}\}.$$

$\mathcal{F}_k$ cannot be empty, since if it were, then $\alpha + 2^{-k}$ would be a strictly greater lower bound on f than α. For each k, let $\mathbf{x}_k$ be some $\mathbf{x} \in \mathcal{F}_k$. $\{\mathbf{x}_k\}_{k=1}^{\infty}$ is a bounded sequence as $\mathcal{F}_k \subseteq \mathcal{F}$, so the Bolzano-Weierstrass theorem we know that there is a convergent subsequence, $\{\mathbf{y}_k\}_{k=1}^{\infty}$, with limit $\bar{\mathbf{y}}$. Because the set is closed, $\bar{\mathbf{y}} \in \mathcal{F}$. By continuity $f(\bar{\mathbf{y}}) = \lim_{k\to\infty} f(\mathbf{y}_k)$, while by construction, $\lim_{k\to\infty} f(\mathbf{y}_k) = \alpha$.

Thus, the optimal solution is $\bar{\mathbf{y}}$. □

4 Convex Optimization

Before we conclude, let's briefly discuss one more concept.

Using the same data as the Galton example (see Figure 2), Figure 4 shows the residual sum of squares, $\text{RSS}(a) = \sum_{i=1}^{m}(y_i - ax_i)^2$, as function of the linear coefficient a. We see that the function

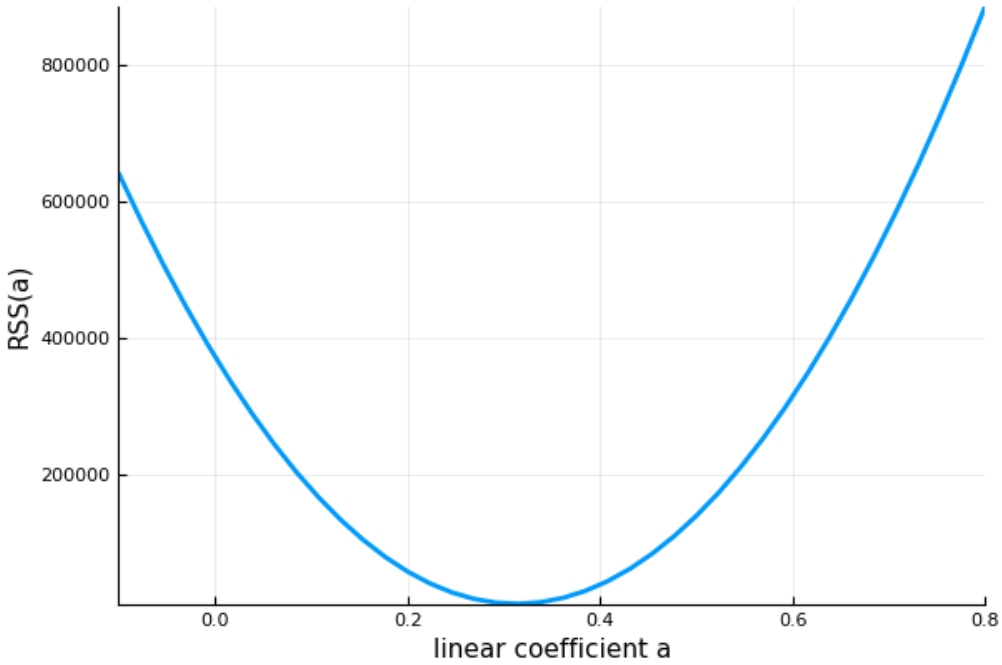

Figure 4: The sum of squared residuals, RSS, as a function of the slope coefficient a, with constant coefficient $b = 46.7$, using the data from the Galton example from Figure 2.

has a special structure: the graph of the function sits below the line joining any two points $(x, f(x))$ and $(z, f(z))$. A function $f : \mathbb{R} \to \mathbb{R}$ that has this property is said to be convex.

Figure 5 shows a convex function, along with two points $(x, f(x))$ and $(z, f(z))$. We see the function sits below the line segment between these points.

Convex functions in more variables are central to optimization, as they form the most important class of objectives for which we can usually develop fast algorithms to solve optimization problems. So why convex functions? In a single sentence: for convex functions a local minimum is also a global minimum, and this fact makes searching for an optimal solution computationally feasible.

Submodular functions are a combinatorial analogue of convex functions for which exact minimization and approximate maximization can be done efficiently.

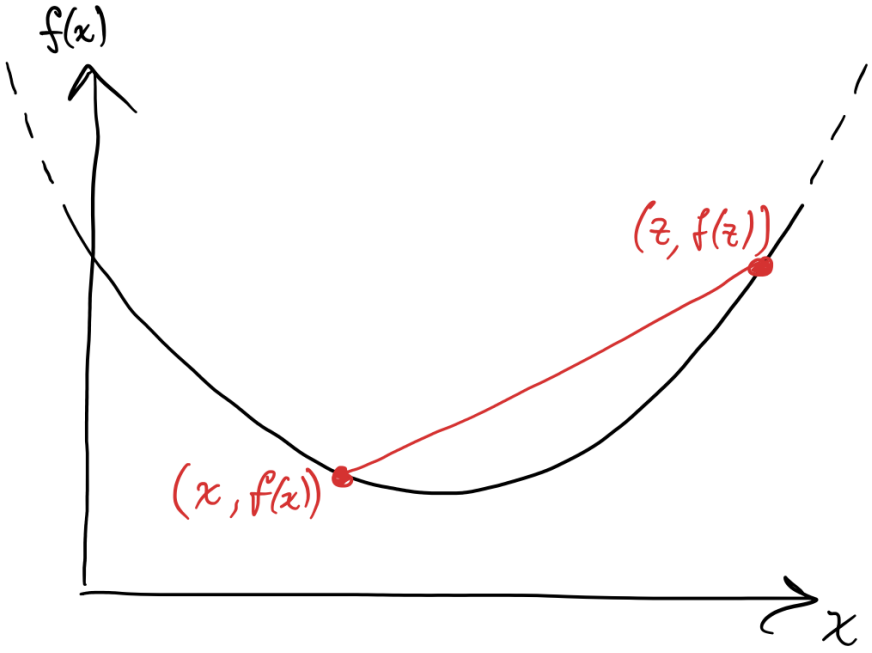

Figure 5: A convex function f.

5 Roadmap

We will begin by introducing basic concepts from convex analysis and prove fundamental properties of convex sets in the next lecture. We will then continue to linear optimization (which is a special case of convex optimization), and cover fundamental concepts like duality, and describe algorithms for solving linear optimization problems. We will then generalize the concepts and develop algorithms for convex optimization problems. We will transition into combinatorial optimization and learn about submodular optimization, a useful discrete analogue of convexity. Finally, we return to convex optimization to touch on some advanced topics.

- Essentials of convex optimization, weeks 1-9
  1. Background, convex analysis
  2. Linear programming
  3. 1st and 2nd order methods
  4. Applications in learning and game theory
  5. Online optimization
- Combinatorial and submodular optimization, weeks 10-11
- Advanced topics in convex optimization, weeks 12-14

6 Remarks

- These notes were adapted from lecture notes by Yaron Singer.
- Francis Galton, in addition to being an accomplished statistician was also a proponent of eugenics. Statistics and optimization have a long history of being used to nefarious ends, alongside their many positive uses throughout science, medicine, engineering, and industry. For example, algorithms are used to make gerrymandering more powerful, while statistical analysis has also been used to argue successfully in courts that gerrymandered districts in North Carolina were unconstitutional. Nonprofit investigative journalism organisation ProPublica has extensive coverage of situations where algorithms may lead to bias. Mitigating unfair bias is an active area of research, see for example this introductory talk on biases of Big Data by Kate Crawford.