The homework assignment is an introduction to the MATLAB software package CVX that will be used in the course. CVX can be downloaded from www.cvxr.com.

We consider the illumination problem of lecture 1. We take $I_{\text{des}} = 1$ and $p_{\text{max}} = 1$, so the problem is

$$
\begin{align*}
\text{minimize} \quad & f_0(p) = \max_{k=1,\ldots,n} |\log(a_k^T p)| \\
\text{subject to} \quad & 0 \leq p_j \leq 1, \quad j = 1, \ldots, m,
\end{align*}
$$

with variable $p \in \mathbb{R}^m$. As mentioned in the lecture, the problem is equivalent to

$$
\begin{align*}
\text{minimize} \quad & \max_{k=1,\ldots,n} h(a_k^T p) \\
\text{subject to} \quad & 0 \leq p_j \leq 1, \quad j = 1, \ldots, m,
\end{align*}
$$

where $h(u) = \max\{u, 1/u\}$ for $u > 0$. The function $h$, shown in the figure below, is nonlinear, nondifferentiable, and convex.

To see the equivalence between (1) and (2), we note that

$$
\begin{align*}
f_0(p) &= \max_{k=1,\ldots,n} |\log(a_k^T p)| \\
&= \max_{k=1,\ldots,n} \max\{\log(a_k^T p), \log(1/a_k^T p)\} \\
&= \log \max_{k=1,\ldots,n} \max\{a_k^T p, 1/a_k^T p\} \\
&= \log \max_{k=1,\ldots,n} h(a_k^T p),
\end{align*}
$$
and since the logarithm is a monotonically increasing function, minimizing $f_0$ is equivalent to minimizing $\max_{k=1,\ldots,n} h(a_k^T p)$.

The specific problem data are given in the file illum_data.m
The command $A = \text{illumdata}$ returns the $n \times m$-matrix $A$ (which has rows $a_k^T$). There are 10 lamps ($m = 10$) and 20 patches ($n = 20$). The coefficients were computed for the geometry shown below, using the formula $a_{kj} = r_{kj}^{-2} \max\{\cos \theta_{kj}, 0\}$ from the lecture.

Use the following methods to compute five approximate solutions and the exact solution, and compare the answers (the vectors $p$ and the corresponding values of $f_0(p)$).

**Equal lamp powers**  Take $p_j = \gamma$ for $j = 1, \ldots, m$. Plot $f_0(p)$ versus $\gamma$ over the interval $[0, 1]$. Graphically determine the optimal value of $\gamma$, and the associated objective value. The objective function $f_0(p)$ can be evaluated in MATLAB as $\max(\text{abs}(\log(A*p)))$.

**Least-squares with saturation**  Solve the least-squares problem

$$\minimize \sum_{k=1}^{n} (a_k^T p - 1)^2 = \|Ap - 1\|^2_2,$$

If the solution has negative coefficients, set them to zero; if some coefficients are greater than 1, set them to 1. Use the MATLAB command $x = A\backslash b$ to solve a least-squares problem (minimize $\|Ax - b\|^2_2$).

**Regularized least-squares**  Solve the regularized least-squares problem

$$\minimize \sum_{k=1}^{n} (a_k^T p - 1)^2 + \rho \sum_{j=1}^{m} (p_j - 0.5)^2 = \|Ap - 1\|^2_2 + \rho \|p - (1/2)1\|^2_2,$$

where $\rho > 0$ is a parameter. Increase $\rho$ until all coefficients of $p$ are in the interval $[0, 1]$. 


Chebyshev approximation  Solve the problem

\[
\begin{align*}
\text{minimize} & \quad \max_{k=1,\ldots,n} |a_k^T p - 1| = \|Ap - 1\|_\infty \\
\text{subject to} & \quad 0 \leq p_j \leq 1, \quad j = 1, \ldots, m.
\end{align*}
\]

We can think of this problem as obtained by approximating the nonlinear function \( h(u) \) by a piecewise-linear function \( |u - 1| + 1 \). As shown in the figure below, this is a good approximation around \( u = 1 \).

This problem can be converted to a linear program and solved using \texttt{linprog}. It can also be solved directly using CVX, with the command \texttt{norm(A*p - 1, inf)} for the cost function.

Piecewise-linear approximation  We can improve the accuracy of the previous method by using a piecewise-linear approximation of \( h \) with more than two segments. To construct a piecewise-linear approximation of \( 1/u \), we take the pointwise maximum of the first-order approximations

\[
h(u) \approx 1/\hat{u} - (1/\hat{u}^2)(u - \hat{u}) = 2/\hat{u} - u/\hat{u}^2,
\]

at a number of different points \( \hat{u} \). This is shown below, for \( \hat{u} = 0.5, 0.8, 1 \). In other words,

\[
h_{\text{pw1}}(u) = \max\{u, \frac{2}{0.5} - \frac{1}{0.5^2}u, \frac{2}{0.8} - \frac{1}{0.8^2}u, 2 - u\}.
\]
Solve the problem

$$\begin{align*}
\text{minimize} & \quad \max_{k=1,\ldots,n} h_{\text{pw1}}(a_k^T p) \\
\text{subject to} & \quad 0 \leq p_j \leq 1, \quad j = 1, \ldots, m
\end{align*}$$

using \texttt{linprog} or CVX.

**Exact solution**  Finally, use CVX to solve

$$\begin{align*}
\text{minimize} & \quad \max_{k=1,\ldots,n} \max(a_k^T p, 1/a_k^T p) \\
\text{subject to} & \quad 0 \leq p_j \leq 1, \quad j = 1, \ldots, m.
\end{align*}$$

Use the function \texttt{inv_pos()} to expresss the function $f(x) = 1/x$ with domain $\mathbb{R}_{++}$. 

\textbf{4}