3.15:

(a) In the limit, we use l'Hospital's rule since numerator and denominator go to zero:

$$\lim_{\alpha\to 0} U_\alpha(x) = \lim_{\alpha\to 0} \frac{x^\alpha - 1}{\alpha} = \lim_{\alpha\to 0} \frac{\frac{d}{d\alpha}(x^\alpha - 1)}{\frac{d}{d\alpha}\alpha} = \lim_{\alpha\to 0} \frac{x^\alpha \log x}{1} = \log x = U_0(x)$$

(b) $U_\alpha(1) = \frac{1^\alpha - 1}{\alpha} = 0.$ $\quad U'_\alpha(x) = x^{\alpha-1} > 0$, so $U_\alpha(x)$ is monotone increasing.

$U''_\alpha(x) = (\alpha - 1)\, x^{\alpha-2} \leq 0$ since $0 < \alpha \leq 1$, which implies $U_\alpha(x)$ is concave.

3.25.

Let $\tilde{C} = \{1, \dots, n\} / C$, which satisfies $\mathrm{prob}(p, C) + \mathrm{prob}(p, \tilde{C}) = 1$ and $\mathrm{prob}(q, C) + \mathrm{prob}(q, \tilde{C}) = 1$. Then we have $\mathrm{prob}(p, C) - \mathrm{prob}(q, C) = -(\mathrm{prob}(p, \tilde{C}) - \mathrm{prob}(q, \tilde{C}))$.

We can rewrite $d_{m,p}$ as: $d_{mp}(p, q) = \max\{\mathrm{prob}(p, C) - \mathrm{prob}(q, C) \mid C \subseteq \{1, \dots, n\}\}$.

Since it is the maximum of 2^n linear functions of (p, q), the $d_{m,p}$ is convex.

Suppose the solution subset C^* is: $C^* = \{i \in \{1, \dots, n\} \mid p_i > q_i\}$.

The indices for which $p_i = q_i$ don't matter to $d_{m,p}$, so we can ignore them. Without loss of generality, for each index, we have $p_i > q_i$ or $p_i < q_i$.

Suppose there is another subset C. If there is an element "e" in C^* but not in C, then by adding "e" to C we can increase $\mathrm{prob}(p, C) - \mathrm{prob}(q, C)$ by $p_e - q_e > 0$. This means that C can not be optimal. On the other hand, if there is an element "e" in C but not in C^*, which implies that $p_e - q_e < 0$. If we delete "e" from C, we then increase $\mathrm{prob}(p, C) - \mathrm{prob}(q, C)$ by $-(p_e - q_e) > 0$. This means C can not be optimal.

Therefore, we have $d_{mp}(p, q) = \sum_{p_i > q_i} (p_i - q_i)$. By definition, we have $1^T p = 1^T q = 1$, we have $\sum_{p_i > q_i}(p_i - q_i) + \sum_{p_i \leq q_i}(p_i - q_i) = 1^T p - 1^T q = 0$, which implies $\sum_{p_i > q_i}(p_i - q_i) = -\left(\sum_{p_i \leq q_i}(p_i - q_i)\right)$

Hence, we have:

$$d_{mp}(p, q) = \sum_{p_i > q_i}(p_i - q_i) = \frac{1}{2}\sum_{p_i > q_i}(p_i - q_i) + \frac{1}{2}\sum_{p_i > q_i}(p_i - q_i)$$

$$= \frac{1}{2}\sum_{p_i > q_i}(p_i - q_i) - \frac{1}{2}\sum_{p_i \leq q_i}(p_i - q_i)$$

$$= \frac{1}{2}\sum_{i=1}^{n} |p_i - q_i| = \frac{1}{2}\|p - q\|_1.$$

3.55:

(a) $f(x) = \int_{-\infty}^{x} e^{-h(t)}\,dt$, $f'(x) = e^{-h(x)}$, $f''(x) = -h'(x)\,e^{-h(x)}$

It is obvious that $f''(x)f(x) = (-h'(x))\,e^{-h(x)}\int_{-\infty}^{x} e^{-h(t)}dt \le 0$ when $h'(x) \ge 0$

Hence , $(-h'(x))\,e^{-h(x)}\int_{-\infty}^{x} e^{-h(t)}dt \le \left(e^{-h(x)}\right)^2 \Leftrightarrow f''(x)f(x) \le (f'(x))^2$

(b) $h(t) \ge h(x) + h'(x)(t-x) \Leftrightarrow -h(t) \le -h(x) - h'(x)(t-x)$

$$\Leftrightarrow e^{-h(t)} \le e^{-h(x) + xh'(x) - th'(x)}$$

$$\Leftrightarrow \int_{-\infty}^{x} e^{-h(t)}dt \le e^{-h(x)+xh'(x)} \int_{-\infty}^{x} e^{-th'(x)}dt$$

$$\Leftrightarrow \int_{-\infty}^{x} e^{-h(t)}dt \le e^{-h(x)+xh'(x)} \cdot \frac{e^{-xh'(x)}}{-h'(x)}$$

$$\Leftrightarrow \int_{-\infty}^{x} e^{-h(t)}dt \le \frac{e^{-h(x)}}{-h'(x)} \quad \text{—— ①}$$

By eq. ①, we have : $\int_{-\infty}^{x} e^{-h(t)}dt \le \frac{e^{-h(x)}}{-h'(x)}$

$$\Leftrightarrow (-h'(x))\int_{-\infty}^{x} e^{-h(t)}dt \le e^{-h(x)} \quad \text{since } h'(x) < 0$$

$$\Leftrightarrow (-h'(x))\,e^{-h(x)}\int_{-\infty}^{x} e^{-h(t)}dt \le \left(e^{-h(x)}\right)^2$$

$$\Leftrightarrow f''(x)\,f(x) \le (f'(x))^2 \quad \text{if } h'(x) < 0$$

A2.21:

(a) Suppose S is expressed as a convex combination of permutation matrices: $S = \sum_k \theta_k P_k$ where $0 \le \theta_k \le 1$, $\sum \theta_k = 1$; and P_k permutation matrices. Since f is convex and symmetric, we have $f(Sx) = f\left(\sum_k \theta_k P_k x\right) \le \sum_k \theta_k f(P_k x) = \sum_k \theta_k f(x) = f(x)$.

(b)
$$Y = Q\,\mathrm{diag}(\lambda)\,Q^T = \begin{bmatrix} Q_{11} & Q_{12} & \cdots & Q_{1n} \\ Q_{21} & & & \\ \vdots & & \ddots & \\ Q_{n1} & & & Q_{nn} \end{bmatrix} \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & & \ddots & \\ 0 & & & \lambda_n \end{bmatrix} \begin{bmatrix} Q_{11} & Q_{21} & \cdots & Q_{n1} \\ Q_{12} & & & \\ \vdots & & \ddots & \\ Q_{1n} & & & Q_{nn} \end{bmatrix}$$

$$\Leftrightarrow \quad Y_{ii} = \sum_{j=1}^{n} Q_{ij}^2 \lambda_j$$

$$\Leftrightarrow \mathrm{diag}(Y) = \begin{bmatrix} \sum_{j=1}^{n} Q_{1j}^2 \lambda_j \\ \sum_{j=1}^{n} Q_{2j}^2 \lambda_j \\ \vdots \\ \sum_{j=1}^{n} Q_{nj}^2 \lambda_j \end{bmatrix} = \begin{bmatrix} Q_{11}^2 & Q_{12}^2 & \cdots & Q_{1n}^2 \\ Q_{21}^2 & & & \\ \vdots & & \ddots & \\ Q_{n1}^2 & & & Q_{nn}^2 \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_n \end{bmatrix} = S\lambda$$

Since $QQ^T = I$, we have $\sum_{j=1}^{n} Q_{ij}^2 = 1$. Since $Q^TQ = I$, we have $\sum_{i=1}^{n} Q_{ij}^2 = 1$

Hence, S is doubly stochastic.

(c) From (a) and (b), we have $X = Q\,\mathrm{diag}(\lambda)\,Q^T$ and $\mathrm{diag}(X) = S\lambda$. Then we conclude that for any symmetric X, we have $f(\mathrm{diag}(X)) = f(S\lambda(X)) \le f(\lambda(X))$.
If V is orthogonal, then $\lambda(X) = \lambda(V^TXV)$. Therefore, also
$f(\mathrm{diag}(V^TXV)) \le f(\lambda(X))$ for all orthogonal V, with equality if $V = Q$.
Hence, $f(\mathrm{diag}(V^TXV)) = \sup_{V \in \mathcal{V}} f(\mathrm{diag}(V^TXV))$. This shows that $f(\lambda(X))$ is convex since it is the supremum of a family of convex functions of X.

A12.1:

(a) We can write the problem as:

minimize: α

subject to: $f_1(a) = \sup_{0 \le \omega \le \frac{\pi}{3}} H(\omega) \le 1.12$

$$f_2(a) = \inf_{0 \le \omega \le \frac{\pi}{3}} H(\omega) \ge 0.89$$

$$f_3(a) = \sup_{\omega_c \le \omega \le \pi} H(\omega) \le \alpha$$

$$f_4(a) = \inf_{\omega_c \le \omega \le \pi} H(\omega) \ge -\alpha$$

For each ω, $H(\omega)$ is a linear function of a. Hence, f_1 and f_3 are convex, f_2 and f_4 are concave. Then it is a convex optimization problem.

(b) We can write the problem as:

minimize $f_5(a)$

Subject to: $f_1(a) = \sup_{0 \le \omega \le \frac{\pi}{3}} H(\omega) \le 1.12$

$$f_2(a) = \inf_{0 \le \omega \le \frac{\pi}{3}} H(\omega) \ge 0.89$$

where $f_5(a) = \inf\{\Omega \mid -\alpha \le H(\omega) \le \alpha \text{ for } \Omega \le \omega \le \pi\}$

We know f_1 is convex, f_2 is concave.

And the sublevel sets of f_5 are:

$\{a \mid f_5(a) \le \Omega\} = \{a \mid -\alpha \le H(\omega) \le \alpha \text{ for } \Omega \le \omega \le \pi\}$ is the intersection of infinite halfspaces, so f_5 is quasiconvex. Therefore, it is a quasiconvex optimization problem.

(c) We can write the problem as:

minimize $f_6(a)$

subject to: $f_1(a) = \sup_{0 \le \omega \le \frac{\pi}{3}} H(\omega) \le 1.12$

$$f_2(a) = \inf_{0 \le \omega \le \frac{\pi}{3}} H(\omega) \ge 0.89$$

$$f_3(a) = \sup_{\omega_c \le \omega \le \pi} H(\omega) \le \alpha$$

$$f_4(a) = \inf_{\omega_c \le \omega \le \pi} H(\omega) \ge -\alpha$$

where $f_6(a) = \min\{k \mid a_{k+1} = \cdots = a_N = 0\}$. We know f_1 and f_3 are convex, f_2 and f_4 are concave. The sublevel sets of f_6 are affine sets:

$\{a \mid f_6(a) \le k\} = \{a \mid a_{k+1} = \cdots = a_N = 0\}$. This means that f_6 is a quasiconvex function. Hence, it is a quasiconvex optimization problem.

A16.1:

(a) In order to get the optimal generators and line power flows, we solve the linear programming:

$$\text{minimize}: \quad c^T g$$

$$\text{subject to}: \quad Ap = \begin{bmatrix} -g \\ d \end{bmatrix}, \quad -P^{max} \preceq p \preceq P^{max}, \quad 0 \preceq g \preceq G^{max}$$

with variables g and p.

Python code:

```
import cvxpy as cp
from rel_pwr_flow_data import *
p = cp.Variable (m)
g = cp.Variable (k)
cost = cp.Problem (cp.Minimize (c.T * g), \
    [A[:k,:] * p == -g, \
    A [k:,:] * p == np.array (d.T).reshape (-1, ), \
    cp.abs (p)   <= np.array (Pmax.T).reshape (-1, ), \
    g <= np.array (Gmax ).reshape (-1, ), \
    g >= 0 ]). solve ( solver = cp.CVXOPT )
```

Then we get optimal cost $= 44.5945$, $g = \begin{bmatrix} 3 \\ 0 \\ 2.3 \\ 7 \end{bmatrix}$

(b) To handle the additional $N-1$ reliability constraint, we have to introduce a set of power flow vectors for each contingency. Then we solve the linear programming:

$$\begin{aligned}
\text{minimize}: \quad & c^T g \\
\text{subject to}: \quad & A p^{(j)} = \begin{bmatrix} -g \\ d \end{bmatrix}, \quad j = 1, \dots, m \\
& p^{(j)}_j = 0, \quad j = 1, \dots, m \\
& -p^{max} \leq p^{(j)} \leq p^{max}, \quad j = 1, \dots, m \\
& 0 \leq g \leq G^{max}
\end{aligned}$$

python code:

```
P = cp.Variable((m, m))
g = cp.Variable((k, 1))
cost = cp.Problem(cp.Minimize(C.T * g),\
    [A[:k, :] * P == -g * np.ones((1, m)),\
     A[k:, :] * P == d.T * np.ones((1, m)),\
     cp.diag(P) == 0,\
     cp.abs(P) <= Pmax.T * np.ones((1, m)),\
     g <= Gmax,\
     g >= 0 ]).solve(solver = cp.CVXOPT)
```

Then we get the optimal cost $= 56.20$ and $g = \begin{bmatrix} 1.925 \\ 1.863 \\ 4. \\ 4.531 \end{bmatrix}$