3.2

(a)令

$$A=\begin{bmatrix} l_1 & l_2 & l_3 & \cdots & l_{20} \\ m_1 & m_2 & m_3 & \cdots & m_{20} \\ s_1 & s_2 & s_3 & \cdots & s_{20} \end{bmatrix}$$

如果$p,\tilde p$无法识别，那么

$$\begin{aligned} Ap &= A\tilde p \\ A(p-\tilde p) &= 0 \end{aligned}$$

即$p-\tilde p\in\mathcal{N}(A)$。

(b)题目的含义是问，是否存在非负系数a_1,a_2,a_3，使得

$$Ap_{\text{test}}=Ap_{\text{match}}=A\begin{bmatrix} u & v & w\end{bmatrix}\begin{bmatrix} a_1 \\ a_2 \\ a_3\end{bmatrix}$$

显然，这和$A\begin{bmatrix} u & v & w\end{bmatrix}$有关，所以不一定成立。

(c)利用(b)求解即可：

```
%(c)
A = [L_coefficients; M_coefficients; S_coefficients];
b = A * test_light;
B = A * [ R_phosphor; G_phosphor; B_phosphor;]';
coef = B \b;
coef
```

```
coef =
    0.4226
    0.0987
    0.5286
```

(d)Beth正确。令$r_i,\tilde r_i$为两个物体的反射率，p为光谱，如果

$$A\underbrace{\begin{bmatrix} r_1 & 0 & \cdots & 0 \\ 0 & r_2 & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & r_{20}\end{bmatrix}}_{R}p=A\underbrace{\begin{bmatrix} \tilde r_1 & 0 & \cdots & 0 \\ 0 & \tilde r_2 & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & \tilde r_{20}\end{bmatrix}}_{\tilde R}p$$

那么

$$A(R-\tilde{R})p=0$$

这说明要使得等式成立，必然有$p\in\mathcal{N}(A(R-\tilde{R}))$，所以如果$p\notin\mathcal{N}(A(R-\tilde{R}))$，那么该关系并不能成立。

3.3

$$\begin{aligned}\|x-a\| \le \|x-b\| &\Leftrightarrow \\ \|x-a\|^2 \le \|x-b\|^2 &\Leftrightarrow \\ (x-a)^T(x-a) \le (x-b)^T(x-b) &\Leftrightarrow \\ x^Tx-2a^Tx+a^Ta \le x^Tx-2b^Tx+b^Tb &\Leftrightarrow \\ 2(b-a)^Tx \le b^Tb-a^Ta\end{aligned}$$

所以

$$\begin{aligned}c&=2(b-a)\\ d&=b^Tb-a^Ta\end{aligned}$$

3.10

(a)因为二次函数大于等于0恒成立，所以

$$\begin{aligned}\Delta=4b^2-4ac&\le 0\\ b^2&\le ac\\ |b|&\le\sqrt{ac}\end{aligned}$$

等号成立当且仅当

$$|b|=\sqrt{ac}$$

此时存在λ，使得

$$a+2b\lambda+\lambda^2=0$$

(b)

$$\begin{aligned}(v+\lambda w)^T(v+\lambda w)&=\|v+\lambda w\|^2\\ &\ge 0\end{aligned}$$

(c)化简$(v+\lambda w)^T(v+\lambda w)$可得

$$(v+\lambda w)^T(v+\lambda w)=v^Tv+2v^Tw\lambda+w^Tw\lambda^2$$

对该式应用(a)(b)得到

$$\left|v^Tw\right|\le\sqrt{v^Tv}\sqrt{w^Tw}$$

(d)由(a)可知，此时存在λ使得

$$(v+\lambda w)^T(v+\lambda w)=\|v+\lambda w\|^2=0$$

所以存在λ使得

$$v=-\lambda w$$

所以当v,w平行时，等号成立。

3.11

(a)$\mathcal{R}(G)$表示所有可能的y。

(b)$\mathcal{N}(H)$表示使得解码结果为0的编码，特别的，如果$v\in\mathcal{N}(H)$，那么

$$\hat{x}=H\hat{y}=H(Gx+v)=HGx=x$$

(c)题目的要求是，找到H，使得

- 存在G，使得$HG=I_3$
- $He_i=0,i=1,2,3$（一位非0的向量输出为0）

由第二个条件可得H每一列都是0，所以H所有元素全为0，这就与第一个条件矛盾，因此无法构造。

3.16

由定义可得

$$\begin{aligned}\cos(\rho_i)&=\frac{\langle x,p_i\rangle}{\|x\|.\|p_i\|}\\&=\langle x,p_i\rangle\\&=\sum_{j=1}^n x_jp_{i,j}\end{aligned}$$

记

$$P=\begin{bmatrix}p_1^T\\\dots\\p_k^T\end{bmatrix}\in\mathbb{R}^{k\times n},\rho=\begin{bmatrix}\cos(\rho_1)\\\dots\\\cos(\rho_k)\end{bmatrix}\in\mathbb{R}^k$$

那么线性方程组为

$$Px=\rho$$

要使得上述方程对任意ρ有唯一解，那么P列满秩即可，即

$$\text{rank}(P)=n$$

3.17

(a)错误，例如

$$A=\begin{bmatrix}1 & 1\\ 1 & 1\end{bmatrix}$$

(b)错误，例如

$$A=\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix},B=\begin{bmatrix}-1 & 0\\ 0 & -1\end{bmatrix}$$

(c)正确，因为A,B为onto，所以存在A_1,B_1，使得

$$\begin{aligned}AA_1&=I_n\\ BB_1&=I_m\end{aligned}$$

那么

$$\begin{bmatrix}A & C\\ 0 & B\end{bmatrix}\begin{bmatrix}A_1 & 0\\ 0 & B_1\end{bmatrix}=\begin{bmatrix}I_n & 0\\ 0 & I_m\end{bmatrix}=I_{m+n}$$

(d)错误，例如

$$A=B=[1]$$

(e)正确，因为$\begin{bmatrix}A\\ B\end{bmatrix}$为onto，所以该矩阵的行向量线性无关，因此$A$的行向量线性无关，$B$的行向量无关，即$A,B$都是onto

(f)正确，记

$$\begin{bmatrix}A\\ B\end{bmatrix}\triangleq\begin{bmatrix}c_1 & \ldots & c_n\end{bmatrix}=\begin{bmatrix}a_1 & \ldots & a_n\\ b_1 & \ldots & b_n\end{bmatrix}$$

如果

$$\sum_{i=1}^n\alpha_ic_i=0$$

那么

$$\begin{bmatrix}\sum_{i=1}^n\alpha_ia_i\\ \sum_{i=1}^n\alpha_ib_i\end{bmatrix}=0$$

因此

$$\sum_{i=1}^n\alpha_ia_i=0$$

由条件可知A列向量线性无关，所以

$$\alpha_i=0,i=1,\ldots,n$$

因此$\begin{bmatrix} A \\ B \end{bmatrix}$列向量线性无关，即列满秩，因此结论成立。

补充题

1

由仿射函数的定义可得，存在$A\in \mathbb R^{2\times 3},b\in \mathbb R^2$，使得

$$T = AP+b$$

现在的条件为

$$\begin{cases}
T^{(1)} &= AP^{(1)} +b\\
T^{(2)} &= AP^{(2)} +b\\
T^{(3)} &= AP^{(3)} +b\\
T^{(4)} &= AP^{(4)} +b
\end{cases}$$

我们的目标是解出A,b，现在将后面三个式子减去第一个式子得到

$$\begin{cases}
T^{(2)} -T^{(1)} &= A(P^{(2)} -P^{(1)})\\
T^{(3)} -T^{(1)} &= A(P^{(3)} -P^{(1)})\\
T^{(4)} -T^{(1)} &= A(P^{(4)} -P^{(1)})
\end{cases}$$

记

$$\begin{aligned}
\tilde T &= \left[\begin{matrix} T^{(2)} -T^{(1)} & T^{(3)} -T^{(1)} & T^{(3)} -T^{(1)}\end{matrix}\right] \in \mathbb R^{2\times 3}\\
\tilde P &= \left[\begin{matrix} P^{(2)} -P^{(1)} & P^{(3)} -P^{(1)} & P^{(4)} -P^{(1)}\end{matrix}\right] \in \mathbb R^{3\times 3}
\end{aligned}$$

因此上述方程可以合并为

$$\tilde T = A\tilde P$$

如果$\tilde P$可逆，那么

$$A = \tilde T \tilde P^{-1}$$

求解出A之后，带入任意一个式子即可得到b:

$$b = T^{(i)} -AP^{(i)}$$

这部分代码如下:

```
import numpy as np

#数据
P_1 = np.array([10, 10, 10])
P_2 = np.array([100, 10, 10])
P_3 = np.array([10, 100, 10])
P_4 = np.array([10, 10, 100])
T_1 = np.array([27, 29])
```

```
T_2 = np.array([45, 37])
T_3 = np.array([41, 49])
T_4 = np.array([35, 55])

#计算
P = np.c_[P_2-P_1, P_3-P_1, P_4-P_1]
T = np.c_[T_2-T_1, T_3-T_1, T_4-T_1]
A = T.dot(np.linalg.inv(P))
b = T_1 - A.dot(P_1)

print("A =", A)
print("b =", b)
```

```
A = [[0.2        0.15555556 0.08888889]
 [0.08888889 0.22222222 0.28888889]]
b = [22.55555556 23.        ]
```

现在假设

$$P = \begin{bmatrix} p \\ p \\ p \end{bmatrix}$$

那么

$$\begin{aligned} T &= AP + b \\ &= \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix} \begin{bmatrix} p \\ p \\ p \end{bmatrix} + \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \\ &= \begin{bmatrix} (a_{11} + a_{12} + a_{13})p + b_1 \\ (a_{21} + a_{22} + a_{23})p + b_2 \end{bmatrix} \end{aligned}$$

要使得

$$\begin{aligned} T_1 \le T_0 \\ T_2 \le T_0 \end{aligned}$$

那么

$$\begin{aligned} (a_{11} + a_{12} + a_{13})p + b_1 \le T_0 \\ (a_{21} + a_{22} + a_{23})p + b_2 \le T_0 \end{aligned}$$

求解该线性方程组即可，注意该问题中$a_{ij} > 0$，所以

$$p \le \min_{i=1,2} \left\{ \frac{T_0 - b_i}{\sum_{j=1}^{3} a_{ij}} \right\}$$

这部分代码如下：

```
T0 = 70
tmp = (T0 - b) / np.sum(A, axis=1)
pmin = np.min(tmp)
print("p_min =", pmin)
```

```
p_min = 78.33333333333333
```

2

由条件可得

$$\|a-b\| = \eta\|a\|$$

由条件可得

$$\eta_{ba} = \frac{\|a-b\|}{\|b\|} = \eta_{ab}\frac{\|a\|}{\|b\|} \triangleq t\eta_{ab}$$

所以只要计算$t=\frac{\|a\|}{\|b\|}$的范围即可。

由条件可得

$$\begin{aligned}
\eta_{ab}^2 &= \frac{\|a-b\|^2}{\|a\|^2} \\
\eta_{ab}^2\|a\|^2 &= \|a\|^2 - 2a^Tb + \|b\|^2 \\
&\le \|a\|^2 + 2\|a\|.\|b\| + \|b\|^2 \\
&\ge \|a\|^2 - 2\|a\|.\|b\| + \|b\|^2
\end{aligned}$$

即

$$\begin{aligned}
\eta_{ab}^2\|a\|^2 &\le \|a\|^2 + 2\|a\|.\|b\| + \|b\|^2 \\
\eta_{ab}^2\|a\|^2 &\ge \|a\|^2 - 2\|a\|.\|b\| + \|b\|^2 \\
\eta_{ab}^2 t^2 &\le t^2 + 2t + 1 \\
\eta_{ab}^2 t^2 &\ge t^2 - 2t + 1
\end{aligned}$$

求解该方程即可，对应代码如下：

```
import numpy as np

eta_ab = 0.1
a1 = 1 - eta_ab ** 2
b1 = 2
c1 = 1
b2 = -2

def solve(a, b, c):
    delta = b ** 2 - 4 * a * c
```

```python
    x1 = (-b - np.sqrt(delta)) / (2 * a)
    x2 = (-b + np.sqrt(delta)) / (2 * a)

    return x1, x2

t1, t2 = solve(a1, b1, c1)
t3, t4 = solve(a1, b2, c1)
print(t1, t2)
print(t3, t4)

tmin = t3
tmax = t4
eta_bamin = eta_ab * tmin
eta_bamax = eta_ab * tmax
print("eta_ba的最小值为{}".format(eta_bamin))
print("eta_ba的最大值为{}".format(eta_bamax))
```

```
-1.1111111111111112 -0.9090909090909091
0.9090909090909091 1.1111111111111112
eta_ba的最小值为0.09090909090909091
eta_ba的最大值为0.11111111111111112
```

接着求解$\theta = \angle(a, b)$的范围，依然利用定义：

$$\begin{aligned}\eta_{ab}^2 \|a\|^2 &= \|a\|^2 - 2a^T b + \|b\|^2 \iff \\ \eta_{ab}^2 \|a\|^2 &= \|a\|^2 - 2\|a\|.\|b\| \cos(\theta) + \|b\|^2 \iff \\ (1 - \eta_{ab}^2)t^2 - 2\cos(\theta)t + 1 = 0\end{aligned}$$

所以

$$\Delta = 4\cos^2(\theta) - 4(1 - \eta_{ab}^2) \geq 0$$

即

$$\cos(\theta) \geq \sqrt{1 - \eta_{ab}^2} \text{或} \cos(\theta) \leq -\sqrt{1 - \eta_{ab}^2}$$

因为

$$t = \frac{\|a\|}{\|b\|} > 0$$

所以$\cos(\theta) > 0$，即

$$\cos(\theta) \geq \sqrt{1 - \eta_{ab}^2}$$

求解得到：

```python
#求角度
theta_min = 0
theta_max = np.arccos(np.sqrt(1-eta_ab**2))
print("theta的最小值为{}".format(theta_min))
print("theta的最大值为{}".format(theta_max))
```

```
theta的最小值为0
theta的最大值为0.10016742116155969
```

3

利用matlab如下命令即可：

```matlab
rank([F g])==rank(F)
```

依次删除某行的数据，记删除后的矩阵为A_1, y_1，之后判断$[A_1, y_1]$的秩是否和A_1的秩相等即可，如果相等，则出错的位置为删除的行：

```matlab
n =  length(ytilde);
flag = 0;
for i = 1:n
    index = [1:(i-1) (i+1):n];
    A1 = A(index, :);
    y1 = ytilde(index);
    res = (rank([A1 y1])==rank(A1));
    if res == 1
        flag = i;
        break;
    end
end

fprintf("第%d个传感器出错\n", flag);
```

```
第11个传感器出错
```