1.1

$$X = \begin{array}{ccc} x^{(1)} & x^{(2)} & x^{(3)} \end{array}$$
$$X = \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix} \longrightarrow XX^T = \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 0 & 0 \\ -1 & -1 \end{bmatrix} = \begin{bmatrix} 2 & 2 \\ 2 & 2 \end{bmatrix}$$

finding eigen vectors of XX^T
$$\longrightarrow \begin{vmatrix} 2-\lambda & 2 \\ 2 & 2-\lambda \end{vmatrix} = (2-\lambda)^2 - 4 = 0 \longrightarrow \begin{cases} 2-\lambda = 2 \longrightarrow \lambda = 0 \\ 2-\lambda = -2 \longrightarrow \lambda = 4 \end{cases}$$

$$\xrightarrow{\lambda=0} \left[\begin{array}{cc|c} 2 & 2 & 0 \\ 2 & 2 & 0 \end{array}\right] \equiv \left[\begin{array}{cc|c} 1 & 1 & 0 \\ 0 & 0 & 0 \end{array}\right] \longrightarrow \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} -x_2 \\ x_2 \end{bmatrix} \longrightarrow \text{eigen vec} = \begin{bmatrix} -1 \\ 1 \end{bmatrix} \xrightarrow{\text{normalize}} \begin{bmatrix} -\frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} \end{bmatrix} = pc_1$$

$$\xrightarrow{\lambda=4} \left[\begin{array}{cc|c} -2 & 2 & 0 \\ 2 & -2 & 0 \end{array}\right] \equiv \left[\begin{array}{cc|c} 1 & -1 & 0 \\ 0 & 0 & 0 \end{array}\right] \longrightarrow \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} x_2 \\ x_2 \end{bmatrix} \longrightarrow \text{eigen vec} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \xrightarrow{\text{normalize}} \begin{bmatrix} \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} \end{bmatrix} = pc_2$$

دو بردار pc_1, pc_2 مولفه های اصلی هستند

1.2

می دانیم که برای تصویر بردار V روی u داریم: $\text{proj}_u(v) = \frac{\langle u,v \rangle}{\langle u,u \rangle} u$

pc_1

اگر تنها pc_1 وجود داشته باشد ، باید واریانس تصاویر داده ها روی pc_1 را حساب کنیم و داریم :

$$\left.\begin{array}{l} \text{proj}_{pc_1}(x^{(1)}) = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \\ \text{proj}_{pc_1}(x^{(2)}) = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \\ \text{proj}_{pc_1}(x^{(3)}) = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \end{array}\right\} \longrightarrow \text{var} = \frac{1}{3}\left(0^2 + 0^2 + 0^2\right) = \boxed{0}$$

pc_2

به صورت مشابه برای pc_2 داریم :

$$\left.\begin{array}{l} \text{proj}_{pc_2}(x^{(1)}) = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \\ \text{proj}_{pc_2}(x^{(2)}) = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \\ \text{proj}_{pc_2}(x^{(3)}) = \begin{bmatrix} -1 \\ -1 \end{bmatrix} \end{array}\right\} \longrightarrow \text{var} = \frac{1}{3}\left(\sqrt{2}^2 + 0^2 + \sqrt{2}^2\right) = \boxed{\frac{4}{3}}$$

1.3 $\text{میانگین خطای بازسازی} = \frac{1}{n}\sum_{i=1}^{n}\left\| x^{(i)} - proj\left(x^{(i)}\right)\right\|^2$

PC_1 $\quad \frac{1}{3}\left[\|x^{(1)}\|^2 + \|x^{(2)}\|^2 + \|x^{(3)}\|^2\right] = \frac{1}{3}\left(2 + 0 + 2\right) = \boxed{\frac{4}{3}}$

PC_2 $\quad \frac{1}{3}\left[\left\|\begin{bmatrix}0\\0\end{bmatrix}\right\|^2 + \left\|\begin{bmatrix}0\\0\end{bmatrix}\right\|^2 + \left\|\begin{bmatrix}0\\0\end{bmatrix}\right\|^2\right] = \boxed{0}$

2.1

می دانیم که با داشتن شبکه بیزی احتمال شرطی هر نود به شرط پدرانش، از سایر نود ها مستقل است. پس داریم:

$p(F \mid A,S,H,N) = p(F)$ $\qquad p(A \mid F,S,H,N) = p(A)$

$p(S \mid A,F,H,N) = p(S \mid A,F)$

$p(H \mid A,F,S,N) = p(H \mid S)$ $\qquad p(N \mid A,F,S,H) = p(N \mid S)$

2.2

$$p(A=1 \mid F,S,H,N) = \frac{p(A=1,F,S,H,N)}{p(F,S,H,N)}$$

$$= \frac{p(A=1)\cancel{p(F)}\,p(S \mid A=1,F)\,\cancel{p(N \mid S)}\,\cancel{p(H \mid S)}}{\sum_{i=0}^{1} p(A=i)\cancel{p(F)}\,p(S \mid A=i,F)\,\cancel{p(N \mid S)}\,\cancel{p(H \mid S)}} = \frac{p(A=1)\,p(S \mid A=1,F)}{\sum_{i=0}^{1} p(A=i)\,p(S \mid A=i,F)}$$

markov blanket

2.3

پتوی مارکو هر نود شامل فرزندان مستقیم، پدران مستقیم و نود هایی است که با آنها فرزند مشترک داریم

markov blanket $(A) = \{S, F\}$

2.4

همان طور که در قسمت ج نشان دادیم، متغیرهایی که عضو پتوی مارکو نباشند، در مرحله آخر خط می خورند و عبارت نهایی تنها شامل اعضای پتوی مارکو است

2.5

تنها متغیری که ثابت نیست و در مرحله امید ریاضی می تواند تغییر کند A است پس تمامی توزیع هایی که A در آن ها حضور دارد نیز باید دوباره بیشینه شوند. پس جواب توزیع های زیر است:

$p(S=0 \mid A=0, F=0)$ $\qquad p(A=0)$
$p(S=0 \mid A=0, F=1)$ $\qquad p(A=1)$
$p(S=0 \mid A=1, F=0)$
$p(S=0 \mid A=1, F=1)$
$p(S=1 \mid A=0, F=0)$
$p(S=1 \mid A=0, F=1)$
$p(S=1 \mid A=1, F=0)$
$p(S=1 \mid A=1, F=1)$

$\boxed{2.6}$

ستون آخر در حقیقت توزیع $p(A=1)$ را اعلام می کند که در مرحله امید ریاضی قبلی به دست آمده است. باقی ستون ها نیز ثابت اند. بنابراین می توان خواسته های سوال را طبق جدول و به صورت empirical گزارش کرد:

طبق ردیف های 5,4,2,1 می توان گفت

$$p(A=1) = \frac{1+1+0.8+0.4}{5} = 0.64$$

$$p(s=1 \mid F=0, A=1) = \frac{p(s=1, F=0, A=1)}{p(F=0, A=1)} = \frac{\frac{1+0.4}{5}}{\frac{1+1+0.8+0.4}{5}} = \frac{0.28}{0.64} = 0.4375$$

$\boxed{3.1}$

همان طور که در درس دیدیم، در ابعاد بالا با مفهومی تحت عنوان curse of dimensionality روبرو می شویم که شهود ما از distance را خراب می کند. از جمله مصداق های آن می توان به موارد زیر اشاره کرد:

- neighborhoods are no longer local
- points in high dimensional spaces are isolated
- the notion of nearest neighbors vanishes
- ...

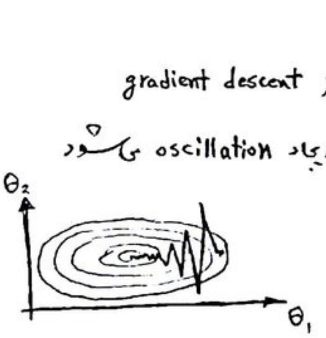
به این معنا که در ابعاد بالا تمام فواصل با هم یکسان می شوند و الگوریتمی مثل k-means دیگر نمی تواند به صورت معنادار و بهینه خوشه بندی را انجام دهد زیرا تمام نقاط فاصله های مشابهی خواهند داشت

$\boxed{3.2}$

در رابطه با تأثیر normalization معمولاً به دو مورد زیر اشاره می شود

1. اگر فیچرها normalize نشده باشند، در محاسبه ی distance، فیچرهایی که scale بزرگتری دارند سایر فیچرها را dominate می کنند. این در حالی است که ممکن است آن فیچرها واقعاً آنقدر مهم نباشند. با normalize کردن تأثیر هر فیچر بر distance نهایی را کنترل می کنیم تا این اتفاق نیفتد

2. اگر فیچرها normalize نشده باشند، converge کردن الگوریتم ها نیز می تواند کندتر شود. به خصوص الگوریتم هایی که از gradient descent استفاده می کنند، اگر scale پارامترهای مدل متفاوت باشد، برداشتن gradient step با اندازه یکسان در این فضا سبب ایجاد oscillation می شود



3.3

آن طور که از مشاهده‌ی داده ها به نظر می رسد، خوشه بندی درست به این صورت است که هر کدام از خم های نعلی شکل باید یک خوشه باشند. اما چون k-means بر اساس فاصله سمپل ها خوشه بندی می کند نمی‌تواند این شکل را capture کند. خوشه بندی نهایی به centroid هایی که در ابتدا انتخاب می شوند نیز وابسته است اما واضح است که مدل ترجیح می دهد نقاط a,c را در یک خوشه بگیرد تا a,b