Assignment 5

STAT 541, Winter 2025

Instructor: Andrew McCormack, mccorma2@ualberta.ca
Due date: April 9th, 11:59 PM.

To receive full marks you must show your work for all derivations.

Problem 1: Clustering

(a) Fix a cluster assignment function $f : \{1, \dots, n\} \to \{1, \dots, k\}$ and let $\hat{\boldsymbol{\mu}}_j = \frac{1}{n_j} \sum_{i:f(i)=j} \mathbf{x}^{(i)}$ be the mean of the observations assigned to the jth cluster by f. n_j is the number of observations assigned to the jth cluster. Show that for every $j = 1, \dots, k$

$$\frac{1}{n_j} \sum_{s:f(s)=j} \sum_{t:f(t)=j} \|\mathbf{x}^{(s)} - \mathbf{x}^{(t)}\|^2 = 2 \sum_{s:f(s)=j} \|\mathbf{x}^{(s)} - \hat{\boldsymbol{\mu}}_j\|^2.$$

Conclude that minimizing the following k-means loss function that we introduced in class (where we plug-in the optimal choice of cluster centers $\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_k$) over f

$$L(f) = \sum_{i=1}^{n} \|\mathbf{x}^{(i)} - \hat{\boldsymbol{\mu}}_{f(i)}\|^2$$

is equivalent to minimizing the following loss function

$$L^*(f) = \sum_{j=1}^{k} \frac{1}{n_j} \sum_{s:f(s)=j} \sum_{t:f(t)=j} \|\mathbf{x}^{(s)} - \mathbf{x}^{(t)}\|^2.$$

The loss function L^* has the intuitive interpretation as the sum across clusters of the average sum squared distances between all points that are assigned to the same cluster. Hence by minimizing L^* we seek a cluster assignment function f such that all points assigned to the same cluster are close to each other.

(b) k-means clustering, and more generally any algorithm that utilizes means, are sensitive to observations that are outliers. Show that if one of the data points, say $\mathbf{x}^{(1)}$ is far away from the remaining data points with $\|\mathbf{x}^{(1)}\| \to \infty$ (leaving the cluster assignment function f all of the other data points fixed) then the cluster center associated with this data point $\hat{\boldsymbol{\mu}}_{f(1)}$ also has a norm that diverges to infinity: $\|\hat{\boldsymbol{\mu}}_{f(1)}\| \to \infty$. (**Hint:** You may find the reverse triangle inequality $\|\mathbf{v} + \mathbf{w}\| \geq \|\mathbf{v}\| - \|\mathbf{w}\|$ useful here.)

Formally, the above limits mean that for every $m > 0$ there exists an $m_* > 0$ such that if $\|\mathbf{x}^{(1)}\| > m_*$ then $\|\hat{\boldsymbol{\mu}}_{f(1)}\| > m$.

Aside: A fix to this is to instead compute medians in the iterations of an analogous clustering algorithm. The geometric median $\hat{\boldsymbol{\theta}}_j$ of the points assigned to class j is defined as

$$\hat{\boldsymbol{\theta}}_j = \underset{\boldsymbol{\theta} \in \mathbb{R}^p}{\operatorname{argmin}} \sum_{i:f(i)=j} \|\mathbf{x}^{(i)} - \boldsymbol{\theta}\|.$$

For geometric medians even if one point within a cluster diverges in norm, as long as the remaining points are fixed, the geometric median will have a bounded norm. Note that for $p \geq 2$ there is generally no closed form expression for a geometric median.

(c) Consider the `Boston` dataset in the package `ISLR2`.

(i) Preprocess the data frame using the `scale` function to ensure that no individual feature has a larger magnitude than the others.

(ii) Using only the first 50 data observations (otherwise the dendrogram becomes too cluttered) and the `dist` function along with `hclust` perform hierarchical clustering with "complete", "single" and "average" linkage.

(iii) Plot each of these fitted models using `medv` as labels in the plot. Given that the data are the average features of houses in a neighborhood roughly what aspects of the neighbourhood do you think the clustering is capturing?

(iv) Suppose you live in the 41st neighbourhood that appears (as a row) in the dataset and are looking to move to a neighbourhood with similar features. According to the complete linkage clustering in (ii) what neighbourhood(s) would you look at?

Problem 2: PCA

For this problem you may assume that the singular values in $\mathbf{D}$ are distinct and ordered so that $d_{11} > \cdots > d_{pp}$.

(a) The sample covariance matrix of a collection of vectors $\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(n)}$ that are independent and identically distributed draws from some distribution P is defined as $S = \frac{1}{n}\sum_{i=1}^{n}(\mathbf{x}^{(i)} - \bar{\mathbf{x}})(\mathbf{x}^{(i)} - \bar{\mathbf{x}})^\top$ (Note sometimes the factor $\frac{1}{n-1}$ appears instead of $\frac{1}{n}$ in the sample covariance). Compute $E(\mathbf{S})$ and compare this to the covariance matrix $\mathbf{\Sigma} \coloneqq \mathrm{Cov}(\mathbf{x}^{(i)}) = E\big(\mathbf{x}^{(i)}(\mathbf{x}^{(i)})^\top\big) - E(\mathbf{x}^{(i)})E(\mathbf{x}^{(i)})^\top$.

(b) Let $\mathbf{v} \in \mathbb{R}^p$ be a unit vector with $\mathbf{v}^\top\mathbf{v} = 1$. Recalling our transformation rules for covariance matrices, the variance of the scalar quantity $\mathbf{v}^\top\mathbf{x}^{(i)}$ (this is the coordinate of the projection of the vector $\mathbf{x}^{(i)}$ onto the line spanned by $\mathbf{v}$) is $\mathbf{v}^\top\mathbf{\Sigma}\mathbf{v}$. Using a Lagrange multiplier for the constraint $\mathbf{v}^\top\mathbf{v} = 1$ find a unit vector $\mathbf{v}$ that maximizes this variance:

$$\underset{\mathbf{v}:\,\mathbf{v}^\top\mathbf{v}=1}{\operatorname{argmax}} \ \mathbf{v}^\top\mathbf{\Sigma}\mathbf{v}$$

(c) Let $\mathbf{X}_c = [\mathbf{x}^{(1)} - \bar{\mathbf{x}}, \cdots, \mathbf{x}^{(n)} - \bar{\mathbf{x}}]^\top \in \mathbb{R}^{n\times p}$ be the the dataframe where each row contains the centered observation $\mathbf{x}^{(i)} - \bar{\mathbf{x}}$. Show that $\mathbf{S} = \frac{1}{n}\mathbf{X}_c^\top\mathbf{X}_c$.

(d) If $\mathbf{X}_c$ has the SVD $\mathbf{X}_c = \mathbf{U}\mathbf{D}\mathbf{V}^\top$ express

$$\hat{\mathbf{v}}_1 = \underset{\mathbf{v}:\,\mathbf{v}^\top\mathbf{v}=1}{\operatorname{argmax}} \ \mathbf{v}^\top\mathbf{S}\mathbf{v}$$

in terms of the left or right singular vectors and values of $\mathbf{X}_c$. How does $\hat{\mathbf{v}}_1$ relate to the principal component directions?

(e) Use the `mvrnorm` function in the `MASS` package to simulate $n = 500$ independent observations from the multivariate normal distribution $\mathcal{N}_3(\mathbf{0}, \mathbf{\Sigma})$ where

$$\mathbf{\Sigma} = \begin{bmatrix} 6 & 2.9 & 0 \\ 2.9 & 3 & 2.9 \\ 0 & 2.9 & 7 \end{bmatrix}$$

Use the `plot3d` function in the `rgl` package to plot these observations in three dimensions. Compute the eigendecomposition of $\mathbf{\Sigma}$ using `eigen`. Explain how this eigendecomposition explains the distribution of points in this plot. How many principal components would appear to be suitable to provide a good reconstruction of this dataset?

Parts (c)-(d)-(e) are bonus questions as we will (hopefully) get to this material only a few days before the assignment due date. If you do not answer (c)-(d)-(e) you can still get 100% on the assignment; if you do answer them correctly this will add 2 points to your grade (out of 25 and capped at 25) for this assignment.

Problem 3: More PCA and Isomap

For this question you will need to load the R packages: `carData, rgl, vegan, vegan3d`.

(a) We examine the `UN98` dataset. Remove the first column of regions from the dataset. Then preprocess the dataset by using `na.omit` to remove countries with missing data and apply `scale` to center and scale the data.

(b) Plot the first two principal component scores of each of the 39 countries along with the names of the country. You should use `svd` to get the PC scores and `text, row.names` for the plotting of country names.

(c) Apply `isomap` to the original scaled dataset of 39 countries using the parameters $ndim = 3$ and $k = 4$. Use `rgl.isomap` to plot the isomap embedding of these points into three dimensions and use `orgltext` to plot the country names next to the points. (An alternative for plotting is to use `plot3d` and extract the points from your isomap object).

(d) Which countries that are known to share cultural, geographic, or socioeconomic similarities appear close to each other in the plots?

(e) Plot the points (using `plot`)

$$\begin{aligned} \mathbf{x}^{(1)} &= (1, 0) \\ \mathbf{x}^{(2)} &= (1, 1) \\ \mathbf{x}^{(3)} &= (0, 1) \\ \mathbf{x}^{(4)} &= (-1, 1) \\ \mathbf{x}^{(5)} &= (-1, 0) \\ \mathbf{x}^{(6)} &= (-1, -1) \\ \mathbf{x}^{(7)} &= (0, -1) \\ \mathbf{x}^{(8)} &= (1, -1) \end{aligned}$$

and perform isomap on these point with $dim = 2$ and $k = 2$ (do not do any rescaling). Is the isomap embedding able to reconstruct the data points exactly? Provide an explanation for what is occurring and why the embedded points take on this particular shape.