STA 35C Statistical Data Science III

Practice Midterm 2 Solution

Instructor: Dogyoon Song

Problem 1: (16 points total). Multiple-choice

(a) Probability, conditional probability, Bayes' rule

- ■ If events A and B are independent, then $\Pr(A \mid B) = \Pr(A)$.
- ■ Bayes' rule states $\Pr(A \mid B) = \dfrac{\Pr(B \mid A)\ \Pr(A)}{\Pr(B)}$.
- □ For disjoint events A and B, $\Pr(A \cap B) = \Pr(A)\ \Pr(B)$.
- □ $\Pr(A \cup B) = \Pr(A) + \Pr(B)$, regardless of whether A and B are disjoint.

Explanations:

- The first two statements are **correct**. If A and B are independent, conditional probability equals $\Pr(A)$. Also, Bayes' rule is exactly $\Pr(A \mid B) = \frac{\Pr(B|A)\,\Pr(A)}{\Pr(B)}$.
- The third is **incorrect**, because for disjoint events $\Pr(A \cap B) = 0 \neq \Pr(A)\ \Pr(B)$ in general (unless one is zero).
- The fourth is **incorrect**, because $\Pr(A \cup B) = \Pr(A) + \Pr(B) - \Pr(A \cap B)$; you only sum them directly if A and B are disjoint.

(b) Linear regression: R^2

- ■ R^2 measures the proportion of variability in Y explained by the model.
- □ R^2 can take on values between -1 and 1.
- ■ If $R^2 = 1$, then the model's fitted values match the actual response perfectly.
- □ A large R^2 always guarantees excellent out-of-sample performance.

Explanations:

- R^2 measures the fraction of Y's variance explained by the model, and $R^2 = 1$ implies a perfect fit on the training data.
- $R^2 = 1 - \frac{\text{RSS}}{\text{TSS}}$, and thus[1] takes value in $[0, 1]$ because $0 \leq \text{RSS} \leq \text{TSS}$.
- A large R^2 does *not* guarantee good generalization, because the model could be overfitted.

[1] This is out of the scope of this course, but R^2 can indeed be outside $[0, 1]$ under certain circumstances (e.g., no intercept or adjusted R^2). Anyways, saying it's between -1 and 1 is imprecise.

(c) Classification thresholds & errors

- ☐ Increasing the decision threshold p^* typically increases the false positive rate.
- ☐ A false negative occurs when the actual class is 0 but we predict 1.
- ☐ Setting $p^* = 0.5$ always optimizes both sensitivity and specificity.
- ■ Lowering p^* generally increases the number of predicted positives.

Explanations:

- Increasing p^* typically *reduces* false positives (so that option is incorrect).
- A false negative occurs if the true class is 1 but we predict 0 (so that statement is incorrect).
- $p^* = 0.5$ is a common default threshold but doesn't guarantee an optimal tradeoff for all problems (so that is incorrect).
- Lowering p^* indeed predicts '1' more frequently, increasing predicted positives.

(d) Best subset selection

- ■ Best subset selection fits *all* possible subsets of predictors of each size to find the best subset.
- ☐ It always returns the model with the lowest test error among all subsets.
- ■ It can suffer from high computational cost if the number of predictors is large.
- ■ It may choose a different best model size than forward stepwise selection.

Explanations:

- By definition, best subset selection enumerates all subsets at each size.
- It identifies the best *training*-fit subset, not necessarily the lowest *test* error (unless we do a separate test to confirm). Even if we choose the best subset using cross-validation, lowest validation error does not necessarily guarantee lowest test error.
- It can be expensive for large p (since 2^p subsets).
- It might produce a different best model size than forward stepwise, since stepwise is a greedy procedure.

(e) Resampling methods: Bootstrap

- ☐ Typically the bootstrap procedure repeatedly samples *without replacement* from the original data.
- ☐ In each bootstrap sample of size n, all original observations must appear at least once.
- ■ The bootstrap can be used to estimate the uncertainty (standard error) of an estimator.
- ☐ A single bootstrap sample is guaranteed to provide lower variance than the original sample.

Explanations:

- Bootstrap sampling is done *with replacement* to imitate i.i.d. sampling, not without.
- Not all points must appear; some may appear multiple times while some may not appear at all.
- A key use of bootstrap is estimating variability (e.g., standard errors, confidence intervals).
- A single bootstrap sample does *not* guarantee any variance reduction.

(f) Regularization: Lasso

- ■ Lasso uses an ℓ_1-penalty on the regression coefficients and can set some coefficients exactly to zero.
- ■ Increasing the penalty parameter λ shrinks coefficients toward zero.
- □ Lasso always provides lower test error than ordinary least squares.
- ■ If $\lambda = 0$, Lasso reduces to the usual least squares solution.

Explanations:

- Lasso uses the ℓ_1 penalty and can drive some coefficients exactly to zero.
- Higher λ means stronger shrinkage.
- Lasso does *not* always outperform OLS (it often helps, but not guaranteed).
- If $\lambda = 0$, it is identical to least squares.

(g) Regression splines: Degree-3 spline

- ■ A degree-3 spline is a piecewise cubic function.
- ■ It is continuous up to its second derivative at each knot.
- □ It must have a continuous third derivative at each knot.
- ■ "Natural splines" are degree-3 splines with additional boundary constraints.

Explanations:

- A degree-3 spline is piecewise cubic with continuity in up to the second derivative at each knot.
- We do *not* require continuity of the third derivative.
- Natural splines have extra boundary constraints that reduce spurious wiggles at the extremes.

(h) Clustering

- □ Clustering is a supervised learning task.
- □ In clustering, each observation has a numeric response Y used to form clusters.
- ■ k-means and hierarchical clustering are two common approaches.
- □ Clustering can only be done with at most two features per observation.

Explanations:

- Clustering is *unsupervised*, so the first is false.
- We do not rely on a numeric response for clustering (it's unlabeled).
- k-means and hierarchical are indeed two standard clustering approaches.
- Clustering can be done in higher dimensions as well (not just 2D).

Problem 2: (24 points). True/False with Justification

(a) "If the first principal component accounts for most of the variation in the predictor variables X, then it must also be the single best predictor of the response Y."

False. The first principal component maximizes *variance in* X, not necessarily correlation with Y or predictive ability for Y.

(b) "When using cross-validation, having more folds (e.g., $k = 10$ instead of $k = 5$) always guarantees a lower test error."

False. While more folds can reduce bias of the CV estimate, it does not guarantee a lower *actual* test error or that it always outperforms fewer folds.

(c) "In Lasso regression, all highly correlated predictors are shrunk equally toward zero."

False. Lasso can treat correlated predictors differently, often zeroing out one while retaining another, unlike Ridge which tends to shrink correlated predictors more uniformly.

(d) "Lowering the decision threshold p^* in a logistic model will generally increase both the false positive rate and the true positive rate."

True. With a lower threshold, we label more observations as class 1, thus catching more true positives but also increasing false positives.

(e) "A high training R^2 guarantees that the model will also generalize well to new data."

False. Overfitting can inflate training R^2 without ensuring good out-of-sample performance.

(f) "The main purpose of the bootstrap is to reduce the bias of an estimator, rather than to assess variability or build confidence intervals."

False. The key purpose of the bootstrap is typically to estimate variability (e.g., standard errors, CIs). Bias can be addressed in some cases, but it's not the main goal.

(g) "When performing clustering, labeled data are critical for computing within-cluster variance."

False. Clustering is an *unsupervised* method that does not require labels; the variance measure uses just the feature values.

(h) "Benjamini-Hochberg procedure can control the family-wise error rate (FWER) when testing multiple hypotheses."

False. Benjamini-Hochberg controls the *false discovery rate (FDR)*, not the FWER.

Problem 3: (20 points total). Statistical Learning

(a) (8 points). Supervised vs. Unsupervised Learning

- **Supervised learning** uses labeled data (X, Y) to build a predictive or explanatory model; e.g. linear regression to predict house prices.
- **Unsupervised learning** uses unlabeled data (X) to discover structure, e.g. clustering customers. The main difference is the presence or absence of labeled responses.

(b) (6 points). Training Error vs. Test Error

- **Training error** measures fit on the *training data*, while **test error** measures out-of-sample generalization on the new, unseen *test data*.
- In the figure, typically the **red curve** is the test error, which first decreases as the model becomes more flexible, then eventually increases due to overfitting.
- The **bias-variance tradeoff** says more flexible models reduce bias but increase variance, and there's an optimal point balancing them.

(c) (6 points). Cross-Validation

- Cross-validation partitions the *training* data into folds, using one fold at a time as a validation set and averaging performance.
- By treating the held-out portion of training data as a surrogate test data, cross-validation simulates how the model might perform on new, unseen data and provides an estimate of test error.
- It assumes the training data are representative of the broader data distribution, so that left-out folds approximate a test set.

Problem 4: (40 points total). Regression

(a) (10 points).

(i) **(5 points) Least squares for** $Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2$ typically means minimizing

$$\sum_{i=1}^{n} \big(Y_i - \beta_0 - \beta_1 X_{i,1} - \beta_2 X_{i,2}\big)^2.$$

In practice, we solve $\frac{\partial}{\partial \beta_j} = 0$ for $j = 0, 1, 2$, or interpret visually as finding the plane that best fits (X_1, X_2, Y) in 3D space, which minimizes the sum of squared residuals (=vertical deviation of the datapoints from the plane).

(ii) **(5 points)** With $\hat{\beta}_0 = -3,\ \hat{\beta}_1 = 5,\ \hat{\beta}_2 = 2$, predicting at $(1, 2)$:

$$\hat{Y} = -3 + 5 \times 1 + 2 \times 2 = -3 + 5 + 4 = 6.$$

(b) (10 points).

(i) **(5 points) Interpret** β_1 in multiple regression as the *average effect* of X_1 on Y while *holding* X_2 *fixed*. In the simpler model $Y = \beta_0 + \beta_1 X_1$, β_1 is just the overall slope of X_1 without controlling for other variables.

(ii) **(5 points) Multiple vs. simple regression interpretation** may differ if X_1 and X_2 are correlated. If X_2 strongly influences Y and is correlated with X_1, omitting X_2 can create a confounding effect. In a binary X_2 scenario (treatment vs. control) illustrated in the figure, controlling for X_2yields separate lines for each group with a negative slope; however, ignoring X_2 mixes the two groups and might distort the slope of X_1 in a simple regression model to be positive. This can happen because the influence of X_2 on Y is marginalized and is falsely attributed to X_1 through the chain $X_1 - X_2 - Y$. In contrast, β_1 in multiple regression only captures the *direct* effect of X_1 on Y, holding X_2 fixed.

(c) (10 points).

(i) **(5 points) Polynomial terms in X_1:** They let the model capture nonlinearity in X_1's relationship with Y, beyond a single straight slope.

(ii) **(5 points) Linear vs. cubic fit:** A straight line may underfit if the data curve, while a degree-3 polynomial can bend to fit more complex patterns. Generally, cubic fits reduce bias (=increase flexibility) but can increase variance if data are limited, so there is a bias-variance tradeoff.

(d) (10 points).

(i) **(5 points) Degree-3 spline constraints:** at each knot c_j, ensure:

- continuity of the function itself:

$$\alpha_0 + \alpha_1 c_1 + \alpha_2 c_1^2 + \alpha_3 c_1^3 = \beta_0 + \beta_1 c_1 + \beta_2 c_1^2 + \beta_3 c_1^3,$$
$$\beta_0 + \beta_1 c_2 + \beta_2 c_2^2 + \beta_3 c_2^3 = \gamma_0 + \gamma_1 c_2 + \gamma_2 c_2^2 + \gamma_3 c_2^3,$$
$$\alpha_4 = \beta_4 = \gamma_4.$$

- continuity of the first derivative:

$$\alpha_1 + 2\alpha_2 c_1 + 3\alpha_3 c_1^2 = \beta_1 c_1 + 2\beta_2 c_1 + 3\beta_3 c_1^2,$$
$$\beta_1 + 2\beta_2 c_2 + 3\beta_3 c_2^2 = \gamma_1 + 2\gamma_2 c_2 + 3\gamma_3 c_2^2.$$

- continuity of the second derivative:

$$2\alpha_2 + 6\alpha_3 c_1 = \beta_2 + 6\beta_3 c_1,$$
$$2\beta_2 + 6\beta_3 c_2 = \gamma_2 + 6\gamma_3 c_2.$$

(ii) **(5 points) Natural cubic spline vs. single degree-3 polynomial:** A single polynomial can behave poorly far from the data center, whereas a natural spline imposes boundary constraints that keep the function more stable at the edges, reducing erratic extrapolation and sometimes lowering degrees of freedom. However, in the middle (where data are more abundant), a natural cubic spline retains similar, or even higher, flexibility to a degree-3 polynomial.

Problem 5: (40 points + 3 bonus). Classification

(a) (10 points). Two-Dimensional Logistic Regression

(i) **(5 points)** We have

$$\log\Big(\frac{p(x)}{1-p(x)}\Big) = -2 - 1\,x_1 + 2\,x_2.$$

For $x_{\text{test}} = (1, 1)$, the linear predictor (log odds) is

$$\eta = -2 - 1 \times 1 + 2 \times 1 = -2 - 1 + 2 = -1.$$

Thus $\hat{p} = \frac{e^{-1}}{1+e^{-1}} \approx 0.269$. Since $0.269 < 0.5$, we predict class 0.

(ii) **(5 points)** The **decision boundary** sets $\eta = 0 \implies -2 - x_1 + 2x_2 = 0 \implies x_2 = \frac{x_1}{2} + 1$. Plot a straight line with slope $1/2$, intercept at $x_2 = 1$ when $x_1 = 0$. One side (above line) is $\hat{Y} = 1$, the other side is $\hat{Y} = 0$.

(b) (10 points). LDA in One Dimension

(i) **(5 points) Computing the Discriminant Functions and Predicting at $X = 2$.**

Let Class 0 data be $\{-2, 0, 1, 2\}$ and Class 1 be $\{3, 4, 6\}$. We estimate the *prior* for Class 0 as $\hat{\pi}_0 = \frac{4}{7}$ and for Class 1 as $\hat{\pi}_1 = \frac{3}{7}$.

- **Sample means:**

$$\bar{x}_0 = \frac{-2+0+1+2}{4} = \frac{1}{4} = 0.25, \quad \bar{x}_1 = \frac{3+4+6}{3} = \frac{13}{3} \approx 4.33.$$

- **Pooled variance $\hat{\sigma}^2$:**

$$\hat{\sigma}^2 = \frac{\sum_{x\in\text{Class }0}(x-\bar{x}_0)^2 \; + \; \sum_{x\in\text{Class }1}(x-\bar{x}_1)^2}{n_0 + n_1 - 2}.$$

We have 4 points in Class 0 ($n_0 = 4$) and 3 points in Class 1 ($n_1 = 3$):

$$\text{Class 0: } x - \bar{x}_0 = \{-2.25,\ -0.25,\ 0.75,\ 1.75\}$$
$$\implies \sum_{x\in\text{Class }0} (x-\bar{x}_0)^2 = 5.0625 + 0.0625 + 0.5625 + 3.0625 = 8.75.$$
$$\text{Class 1: } x - \bar{x}_1 = \{-1.33,\ -0.33,\ 1.67\},$$
$$\implies \sum_{x\in\text{Class }1} (x-\bar{x}_0)^2 \approx 1.77 + 0.11 + 2.79 = 4.67.$$
$$\text{Total SSE} \approx 8.75 + 4.67 = 13.42.$$

Hence,

$$\hat{\sigma}^2 = \frac{13.42}{4+3-2} = \frac{13.42}{5} \approx 2.68, \qquad \text{and thus,} \quad \hat{\sigma} \approx 1.64.$$

The **linear discriminant function** in one dimension is often written as

$$\delta_k(x) = \frac{x\,\bar{x}_k - \frac{1}{2}\,\bar{x}_k^2}{\hat{\sigma}^2} + \ln(\hat{\pi}_k),$$

ignoring any constant terms that do not depend on k or x.

Predicting at $x = 2$: Numerically, $\delta_0(2) > \delta_1(2)$; hence LDA predicts **Class 0** at $x = 2$.

****Additional remark:**** Notice that

$$\delta_1(x) - \delta_0(x) = \frac{\bar{x}_1 - \bar{x}_0}{\hat{\sigma}^2}\left(x - \frac{\bar{x}_0 + \bar{x}_1}{2}\right) + \log\left(\frac{\hat{\pi}_1}{\hat{\pi}_0}\right).$$

Since 2 is closer to $\bar{x}_0 = 0.25$ than to $\bar{x}_1 = 4.33$, and $\hat{\pi}_0 > \hat{\pi}_1$, we can see that $\delta_1(x) - \delta_0(x) < 0$ and conclude LDA would predict **Class 0** at $x = 2$, without estimating $\hat{\sigma}$.

(ii) **(5 points) Why LDA is Generative; Explicit PDF Computation.**

LDA assumes each class has a Gaussian distribution with class-specific mean and a common variance $\hat{\sigma}^2$, plus prior $\hat{\pi}_k = \Pr(\text{Class} = k)$. For *one-dimensional* x, the class-k PDF is

$$p(x \mid y = k) = \frac{1}{\sqrt{2\pi\,\hat{\sigma}^2}} \exp\Big(-\frac{(x-\bar{x}_k)^2}{2\,\hat{\sigma}^2}\Big),$$

and each class is chosen with probability $\hat{\pi}_k$. That is,

$$p(x,\, y = k) = \hat{\pi}_k \;\cdot\; p(x \mid y = k).$$

For a new sample $(2, 1)$:

$$p\big(x = 2,\, y = 1\big) = \pi_1 \cdot \frac{1}{\sqrt{2\pi\,\hat{\sigma}^2}} \exp\Big(-\frac{(2 - \bar{x}_1)^2}{2\,\hat{\sigma}^2}\Big).$$

We compare these to the analogous expression for class 0 to classify a new point. Thus, LDA is *generative* because it models how x is generated by first picking a class (with prior $\hat{\pi}_k$), then drawing x from that class's normal distribution.

(c) (10 points). Changing Threshold p^* or Data Imbalance

(i) **(5 points)** If $p^* < 0.5$, we classify '1' more easily. For instance, in medical screening we might want fewer false negatives, so we set $p^* < 0.5$. Conversely, if we want fewer false positives, we raise p^*.

(ii) **(5 points)** Adding three times more data from one class affects both logistic regression (since it updates the log-odds fit) and LDA (since priors and possibly means shift). Both boundaries can move, unless one explicitly fixes prior probabilities or weights classes.

(d) (10 points + 3 bonus). Confusion Matrix and Changing Threshold

(i) **(5 points)** The confusion matrix:

	Pred = 1	Pred = 0
$Y = 1$	45	5
$Y = 0$	10	40

So

$$\text{TP} = 45,\ \text{FN} = 5,\ \text{FP} = 10,\ \text{TN} = 40.$$

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}} = \frac{45}{50} = 0.90, \quad \text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}} = \frac{10}{50} = 0.20.$$

(ii) **(5 points)** Lowering p^* from 0.5 to 0.1 increases the number of predicted positives, so we expect *more false positives* but *fewer false negatives* (since we predict 1 more often).

(iii*) (3 bonus points*) To minimize FPR while keeping TPR $\geq 90\%$, we look at the ROC curve. We choose the threshold corresponding to a point on the ROC curve with TPR ≥ 0.90 but the smallest possible FPR. This is typically found by scanning thresholds until TPR hits 0.90, then picking the threshold that yields the smallest FPR among those points.

Problem 6: (20 points) Inference & Hypothesis Testing

(a) (7 points) Simple Linear Regression Significance

(i) **(4 points)** We have coefficient estimate = 4, std. error = 1.5. Then

$$t = \frac{4}{1.5} \approx 2.67.$$

Checking the table, $t = 2.67$ is between 2.5 and 3.0, so two-sided p-value is between 0.0124 and 0.0027, definitely < 0.05. Thus it is significant.

(ii) **(3 points)** This implies X_1 is significantly associated with Y; i.e., there is strong evidence that $\beta_1 \neq 0$ in this simple linear model.

(b) (7 points) Bootstrapping a Complex Parameter

(i) **(4 points)** Let the original estimate of the parameter be $\hat{T} = 10$. We have six bootstrap estimates $\{7, 8, 10, 11, 12, 15\}$. We compute their sample standard deviation:

$$s_{\text{boot}} = \sqrt{\frac{1}{6-1}\sum_{i=1}^{6}\big(\hat{T}_i - \overline{T}_{\text{boot}}\big)^2} \;\approx\; 2.88.$$

A *normal-approximation* 95% CI is

$$\hat{T} \;\pm\; z_{0.975} \times s_{\text{boot}}, \quad \text{where } z_{0.975} \approx 1.96.$$

Thus

$$95\%\ \text{CI} \approx 10 \;\pm\; 1.96 \times 2.88 \;=\; (4.35,\ 15.65).$$

(ii) **(3 points)** The "95%" means that if we repeated the entire bootstrap process many times, about 95% of such CIs would contain the true parameter T. It is a statement about long-run coverage under repeated sampling from the same data-generating process.

(c) (6 points) Multiple Testing and Bonferroni

(i) **(3 points)** The **family-wise error rate (FWER)** is the probability of making at least one Type I error among the m tests. Symbolically,

$$\text{FWER} = \Pr\big(\text{at least one } H_0 \text{ is true but rejected}\big) = \Pr\big(N_1 \geq 1\big).$$

(ii) **(3 points)** The **Bonferroni correction** sets each individual test's significance level to α/m, ensuring the overall chance of any false rejection is at most α. Hence, FWER $\leq \alpha$.

Problem 7: (20 points). Model Selection & PCA

(a) (8 points) Best Subset vs. PCA

- **Best subset selection**: We have a response Y and p predictors, and we systematically search (or evaluate) all subsets of predictors to find which best fits Y.
- **PCA**: We have a set of features X (potentially high-dimensional), and we find new orthogonal directions of maximal variance. PCA does not use Y at all—it is unsupervised, aiming for dimensionality reduction or feature extraction.
- **Contrast:** Best subset yields a discrete subset of original predictors for modeling Y, whereas PCA yields a lower-dimensional *subspace* spanned by linear combinations of X. BSS is typically chosen via a model-comparison criterion (adjusted R^2, cross-validation etc.), while PCA focuses on maximizing variance in X retained after projection.

(b) (6 points) 2D Dataset and Directional Variance

(i) **(3 points) Directional variance along $\mathbf{v} = (1, 1)$:** For the 2D points $\{(1,2), (2,3), (3,5), (4,6), (5,5)\}$, first compute $(x_i + y_i)$ for each, then scale by $1/\sqrt{2}$. We get:

$$\big\{\tfrac{3}{\sqrt{2}},\ \tfrac{5}{\sqrt{2}},\ \tfrac{8}{\sqrt{2}},\ \tfrac{10}{\sqrt{2}},\ \tfrac{10}{\sqrt{2}}\big\}.$$

Numerically, their sample variance is about 4.85. Hence

$$\text{Var}_{\mathbf{v}}(\mathcal{X}) \approx 4.85.$$

(ii) **(3 points) Comparing to PC1 variance:** The first principal component is the direction that maximizes variance. Therefore, the variance along $\mathbf{v} = (1, 1)$ can be less than or equal to (if $\mathbf{v}$ happens to align with PC1) the variance along PC1, but never strictly guaranteed to exceed it.

(c) (6 points) PCA with 7 variables (variances: 24, 12, 7, 3, 2, 1, 1)

(i) **(3 points)** The total variance is $24 + 12 + 7 + 3 + 2 + 1 + 1 = 50$. Summation of the top 3 principal components is $24 + 12 + 7 = 43$. Hence, the *cumulative proportion* is $43/50 = 0.86$, i.e. 86%.

(ii) **(3 points)** If we keep too few PCs, we risk losing important information. If we keep too many, we might keep noise and hurt interpretability. We typically use a scree plot, trying to find an "elbow" where the slope abruptly changes, suggesting much reduced marginal utility of adding further PCs, or a cumulative PVE to guide the number of components to retain.

Problem 8: (20 points). Clustering

(a) (7 points). Main Ideas of Clustering

(i) **(4 points)** The main goal is to group unlabeled points into clusters so that points within each cluster are "similar" (e.g., smaller pairwise distances in the feature space), while points in different clusters are "dissimilar." This is *unsupervised*, meaning no response labels are used.

(ii) **(3 points) Clustering vs. Classification:** Classification uses labeled data to learn a decision rule for predicting labels of new observations. Clustering uses unlabeled data to discover inherent structure (clusters) in X only.

(b) (7 points). 2-means on $(1, 1)$**,** $(2, 0)$**,** $(8, 6)$**,** $(9, 9)$

(i) **(4 points)** A natural partition places $(1, 1)$ and $(2, 0)$ together, with centroid $\big(1.5,\ 0.5\big)$, and $(8, 6)$ and $(9, 9)$ together, with centroid $\big(8.5,\ 7.5\big)$. Each pair is close in feature space.

(ii) **(3 points)** k-means might not always yield that exact solution because it is a heuristic iterative algorithm, which can get stuck at local minima depending on initialization. Different initial seeds might produce a different partition.

(c) (6 points). Hierarchical Clustering

(i) **(3 points)** With **complete linkage**, we first merge $(1, 1)$ and $(2, 0)$ at distance $\sqrt{2} \approx 1.41$. Next, $(8, 6)$ merges with $(9, 9)$ at distance $\sqrt{10} \approx 3.16$. Finally, these two subclusters merge at the maximum distance among cross-pairs, ≈ 11.40. To form two clusters, we "cut" the dendrogram below height 11.40 (and above $\sqrt{10}$), obtaining the same clusters as above.

(ii) **(3 points) k-means vs. hierarchical:** k-means is typically faster for large n ($O(n)$ vs $O(n^2)$) but requires specifying the number of clusters k in advance, and can converge to local minima. Hierarchical clustering doesn't need k fixed in advance (you can "cut" the dendrogram at various levels) but is more computationally expensive for big n and cannot reverse merges once done.