Midterm 1 Practice: Answer Key

Worked Solutions and Explanations

Q1. Modeling Basics

Item	Select
1.1	A, B, D
1.2	A, D
1.3	E only

1.1. Feature Representation: A, B, D

A. True. Sixty examples give 60 rows; the intercept and two measured features give three columns.

B. True. The input features may be nonlinear functions of the original measurements. The coefficients still enter as a linear combination of fixed features.

C. False. Having at least as many examples as coefficients is necessary for full column rank but does not guarantee it. Feature columns can be dependent; for example, $x_2 = x_1$ creates duplicate columns.

D. True. For every row, the three category indicators sum to one, duplicating the intercept as a linear combination. The complete column set is dependent.

E. False. A, B, and D are true.

Q1. Modeling Basics (continued)

1.2. Cross-Validation: A, D

A. True. Eight configurations times five folds gives 40 fits. Each fit uses 64 training and 16 validation examples; a final refit adds one fit, for 41 total.

B. False. A fold's validation features influence a scaler fitted on all 80 examples. Preprocessing that learns from the data belongs inside each fold even when it does not use labels.

C. False. CV uses different fitted models trained on smaller overlapping subsets. Its score is an estimate, and the minimum is affected by selection. It cannot be guaranteed to equal the population error of the final refit.

D. True. Once the configuration has been selected, fitting on all development data is valid. Test data must use that final fitted scaler and remain outside all fitting and selection.

E. False. A and D are true.

1.3. Regularization: E only

A. False. The gradient equation is

$$\frac{2}{60}X^T(Xw - y) + 2\lambda I'w = 0,$$

so the matrix is $X^TX + 60\lambda I'$. The factor of 60 follows from the stated MSE objective.

B. False. Positive ridge penalties on all slope coefficients give uniqueness even when those columns are dependent, while the single nonzero intercept column identifies w_0. For $v \neq 0$,

$$v^T(X^TX + 60\lambda I')v = \|Xv\|_2^2 + 60\lambda(v_1^2 + v_2^2) > 0.$$

The last inequality holds because either a slope is nonzero or $v_0\mathbf{1} \neq 0$.

C. False. Multiplying the *whole* objective by a positive constant preserves its ordering over all w and therefore its minimizer. Multiplying only the data-fit term would change the relative regularization strength.

D. False. Since $\lambda > 0$, the $\lambda|w_1|$ term creates a kink at $w_1 = 0$. An ordinary partial derivative does not exist there; convexity does not imply differentiability.

E. True. All of A–D are false.

Must/might answers and reasoning

Q2. Polynomial Regression

Must be true. Every function in $\mathcal{H}_1$ is available in $\mathcal{H}_2$ by setting $w_2 = 0$. Minimizing the same training loss over a larger set cannot yield a larger minimum.

What this does not establish. It does not guarantee lower validation or population error. The statement requires the same data, the same unregularized loss, and successful minimization. No assumption that the true function is linear or quadratic is needed.

Q3. Bias and Variance

Might not be true. The stated equality makes squared bias zero, but model variance can remain positive. The expected MSE is $\mathrm{Var}_D(g^{(D)}(x_*)) + \sigma^2$, which exceeds σ^2 whenever that variance is positive.

Concrete counterexample. Suppose the procedure predicts $f(x_*) - 1$ or $f(x_*) + 1$, each with probability $1/2$ across training datasets. Its average prediction is correct, but model variance is 1, so expected MSE is $1 + \sigma^2$.

Formula-sheet connection. The sheet's bias(x) is already a squared difference. Do not square it again. The noise assumption removes cross terms, not the variation of the fitted predictor across datasets.

Interpretation of the response choices. "Might not be true" does not mean that the statement is always false. Q3 equals σ^2 for a perfectly stable unbiased predictor, but the assumptions do not force stability.

Open-ended conceptual answers

Q4. Learning Rate and Feature Scale

4.1. Learning Rate. The learning rate controls convergence speed and stability. A very small rate can make progress slow. Increasing it can accelerate convergence, but an excessive rate can repeatedly overshoot, causing persistent oscillation or divergence. Shrinking oscillations can still converge.

For this convex quadratic, a sufficiently small fixed rate converges to a least-squares minimum. The suitable range depends on the loss curvature; convexity alone does not make every positive rate safe.

Inspect loss and gradient magnitudes across iterations. Sustained progress with gradients approaching zero supports convergence; growing loss or persistent large oscillations suggest instability. A flat loss curve can reflect tiny steps rather than proximity to a minimum, so it is insufficient evidence of convergence on its own.

4.2. Feature Scaling. No. Predictions match because $w_0 + (w_1/c)(cx_1) = w_0 + w_1x_1$. Each original level set is compressed by a factor of c along the slope-coordinate axis. Convexity and the available prediction functions are preserved.

At corresponding weights, residuals match, but the slope gradient contains the feature:

$$\frac{\partial J}{\partial w_1} = \frac{1}{N}\sum_i (\hat{y}^{(i)} - y^{(i)})x_1^{(i)}.$$

The new slope gradient is c times the original; the intercept gradient is unchanged. The slope curvature grows by c^2. The same rate can become too large, though divergence is not inevitable. Matching current predictions does not preserve subsequent updates.

Choose a sufficiently small learning rate for the transformed problem, or standardize the feature using training statistics and select a suitable rate. Adjust coefficients if retaining the current predictor, and apply the same fitted preprocessing to later examples. Check loss and gradients for stability and progress. These changes address optimization while preserving the linear model's available prediction functions.

Q5. Learning Curves

A's training and validation errors settle near 5.2, well above the stated noise floor of 1. Since optimization succeeds, this supports underfitting/high bias. More data of the same kind has had little effect: validation MSE changes from 5.3 to 5.2. Try richer features or a more expressive class, then compare its held-out performance. A is already unregularized, so "reduce its positive penalty" is not an available intervention as stated.

B has strong evidence of high variance at $N = 30$: training MSE is 0.1 while validation MSE is 8.0. As the sample grows, its validation error falls to 1.4 and the gap shrinks. More representative data has helped. At small sample sizes, compare a lower degree or positive ridge/lasso regularization. Additional data may still help at $N = 480$, though improvements may be smaller as risk approaches the noise floor. No procedure can eliminate the fresh-label noise under the given assumptions.

At $N = 480$, A's gap is $5.2 - 5.1 = 0.1$ and B's is $1.4 - 0.9 = 0.5$. B nevertheless has much lower validation MSE, so the colleague's gap-only criterion is wrong. Compare candidate degrees and penalty strengths on the same validation split or CV folds, with preprocessing fitted inside each training fold. Refit the selected procedure on development data, then evaluate on a test set that has not guided those decisions.

Q6. Ridge Regression

6.1. Gradient

Let $e_i(w) = w_0 + w_1 x_1^{(i)} - y^{(i)}$. The factor 2 from differentiating the square cancels the 2 in $3/(2N)$, leaving $3/N$:

$$\boxed{\frac{\partial J_*}{\partial w_0} = \frac{3}{N}\sum_{i=1}^{N} e_i(w)}, \qquad \boxed{\frac{\partial J_*}{\partial w_1} = \frac{3}{N}\sum_{i=1}^{N} e_i(w) x_1^{(i)} + w_1.}$$

The penalty has derivative zero for w_0 and w_1 for w_1. With $I' = \mathrm{diag}(0, 1)$,

$$\boxed{\nabla J_*(w) = \frac{3}{N} X^T (Xw - y) + I'w.}$$

The formula sheet's gradient $X^T(Xw - y)/N$ corresponds to half-MSE, so it needs the factor 3 here, plus the penalty derivative.

6.2. Gradient Descent Update

The augmented matrix, targets, starting predictions, and signed errors are

$$X = \begin{bmatrix} 1 & -1 \\ 1 & 0 \\ 1 & 1 \end{bmatrix}, \quad y = \begin{bmatrix} 0 \\ 1 \\ 3 \end{bmatrix}, \quad Xw^{[0]} = \begin{bmatrix} 2 \\ 1 \\ 0 \end{bmatrix}, \quad e = Xw^{[0]} - y = \begin{bmatrix} 2 \\ 0 \\ -3 \end{bmatrix}.$$

Since $N = 3$,

$$\nabla J_*(w^{[0]}) = \frac{3}{3}\begin{bmatrix} 1 & 1 & 1 \\ -1 & 0 & 1 \end{bmatrix}\begin{bmatrix} 2 \\ 0 \\ -3 \end{bmatrix} + \begin{bmatrix} 0 \\ -1 \end{bmatrix} = \begin{bmatrix} -1 \\ -5 \end{bmatrix} + \begin{bmatrix} 0 \\ -1 \end{bmatrix} = \boxed{\begin{bmatrix} -1 \\ -6 \end{bmatrix}}.$$

Both coordinates use this same old gradient:

$$\boxed{w^{[1]} = \begin{bmatrix} 1 \\ -1 \end{bmatrix} - \frac{1}{6}\begin{bmatrix} -1 \\ -6 \end{bmatrix} = \begin{bmatrix} 7/6 \\ 0 \end{bmatrix}.}$$

Thus the new prediction is the constant 7/6.

Common mistakes. Omitting the factor 3, using $y - Xw$ while retaining the subtraction update, penalizing w_0, or reusing an updated coordinate in the second derivative are distinct conceptual errors.

Q6. Ridge Regression (continued)

6.3. Loss and R^2

Initially, squared errors sum to $4 + 0 + 9 = 13$ and $w_1^2 = 1$, so $J_*(w^{[0]}) = \frac{1}{2}(13) + \frac{1}{2}(1) = 7$. After the update, $w^{[1]} = (7/6, 0)^T$. The errors are

$$Xw^{[1]} - y = \begin{bmatrix} 7/6 \\ 1/6 \\ -11/6 \end{bmatrix}, \qquad \boxed{\mathrm{RSS}(w^{[1]}) = \frac{49 + 1 + 121}{36} = \frac{19}{4}.}$$

The slope penalty is now zero, hence $J_*(w^{[1]}) = \frac{1}{2} \cdot \frac{19}{4} = \frac{19}{8}$. The target mean is $\bar{y} = 4/3$, and

$$\boxed{\mathrm{TSS} = \left(-\frac{4}{3}\right)^2 + \left(-\frac{1}{3}\right)^2 + \left(\frac{5}{3}\right)^2 = \frac{14}{3}.}$$

Therefore

$$\boxed{R^2 = 1 - \frac{19/4}{14/3} = 1 - \frac{57}{56} = -\frac{1}{56}.}$$

The sign can be justified without expanding the squares:

$$\mathrm{RSS}(w^{[1]}) = \mathrm{TSS} + 3\left(\frac{7}{6} - \frac{4}{3}\right)^2 > \mathrm{TSS} \quad \Longrightarrow \quad R^2 < 0.$$

After one step, this predictor has not reached the minimum-RSS fit. An intercept alone does not guarantee nonnegative training R^2.

6.4. Ridge Solution

Factor out 3/2:

$$J_*(w) = \frac{3}{2}\left[\frac{1}{N}\,\mathrm{RSS}(w) + \frac{1}{3}w_1^2\right].$$

Thus $\boxed{c = 3/2}$ and $\boxed{\lambda_{\text{adjusted}} = 1/3}$. The positive outside constant changes neither the minimizer nor the normal equations.

$$X^T X = \begin{bmatrix} 3 & 0 \\ 0 & 2 \end{bmatrix}, \qquad X^T y = \begin{bmatrix} 4 \\ 3 \end{bmatrix}, \qquad N\lambda_{\text{adjusted}} = 1.$$

Therefore

$$\begin{bmatrix} 3 & 0 \\ 0 & 3 \end{bmatrix} w_* = \begin{bmatrix} 4 \\ 3 \end{bmatrix}, \qquad \boxed{w_* = (4/3, 1)^T.}$$

Directly setting the gradient from 6.1 to zero gives the same result. Merely substituting the visible penalty coefficient 1/2 as $\lambda_{\text{adjusted}}$ would ignore the data-loss scaling.

Q7. kNN and Generalization

7.1. Nearest Neighbors

The ordering is

training input	5	8	3	0	-2
distance to 6	1	2	3	6	8
regression target	7	8	4	2	1
class	B	B	A	A	A

There are no distance ties at this query.

k	Regression prediction	Classification prediction
1	7	B
3	$(7+8+4)/3 = 19/3$	B (two votes to one)
5	$(1+2+4+7+8)/5 = 22/5$	A (three votes to two)

7.2. Validation and k

At $x = 1$, the ordered inputs are $0, 3, -2, 5, 8$ with distances $1, 2, 3, 4, 7$. The predictions for $k = 1, 3, 5$ are respectively 2, $(2+4+1)/3 = 7/3$, and $22/5$. The second validation input is $x = 6$, already computed above. Average the two squared errors:

k	Prediction at 1	Prediction at 6	Validation MSE
1	2	7	$[(2-3)^2 + (7-6)^2]/2 = 1$
3	7/3	19/3	$[(-2/3)^2 + (1/3)^2]/2 = 5/18$
5	22/5	22/5	$[(7/5)^2 + (-8/5)^2]/2 = 113/50$

For $k = 3$, both absolute validation errors are below 1; for $k = 1$, both equal 1; and for $k = 5$, both exceed 1. Thus $\boxed{k = 3}$ has the smallest MSE, without needing decimals or expanded squares. Fully substituted MSE expressions are sufficient. The denominator is 2, the validation-set size, not the number of neighbors or training examples.

This comparison does not establish that $k = 3$ is certainly the population-optimal choice; the validation set is very small.

7.3. Training Error

No. The 1-neighbor model memorizes each distinct training input's target. Its zero training MSE does not estimate its error on new examples. The held-out comparison favors $k = 3$.

Q7. kNN and Generalization (continued)

7.4. Hoeffding Bound

(a) Quantities. The observed test error is

$$\boxed{\nu = \frac{24}{200} = \frac{3}{25}, \qquad K = 200, \qquad a = 0, \qquad b = 1.}$$

The population mean μ is the true zero–one error E_{out} of the final classifier. The K on this formula sheet counts evaluation examples; it is not $k = 3$.

(b) Probability guarantee. With tolerance $\varepsilon = 1/10$,

$$P(|\nu - \mu| > 1/10) \le 2\exp\left[-\frac{2(1/10)^2(200)}{(1-0)^2}\right] = \boxed{2e^{-4}}.$$

Therefore the confidence guarantee is at least

$$\boxed{1 - \delta = 1 - 2e^{-4}.}$$

The fully substituted exponential or $2e^{-4}$ is an exact answer; no approximation is needed. This is a bound on the failure probability; it need not equal the actual probability of failure.

(c) Interval. Intersect $[\nu - \varepsilon, \nu + \varepsilon]$ with $[0, 1]$:

$$\boxed{E_{\text{out}} \in \left[\frac{3}{25} - \frac{1}{10}, \frac{3}{25} + \frac{1}{10}\right] = \left[\frac{1}{50}, \frac{11}{50}\right]}$$

with the repeated-sampling guarantee above. The interval already lies inside $[0, 1]$. The guarantee concerns how close the observed error is to the true error. Its confidence level is not the classifier's accuracy. Either interval form is exact.

(d) Selection on the test examples. Choosing k after comparing candidates on these same 200 labels would make the selected classifier depend on the evaluation data. The winning error can be optimistic, so the fixed-classifier premise of this single-model calculation fails. Use an untouched evaluation set, or an appropriate simultaneous-selection correction under its own assumptions.

Topic Guide

Topic	Questions
Dimensions, intercept, encodings, feature maps	1.1; 6.1–6.2
Least squares, nested classes, invertibility	1.1; 2; 6.3–6.4
Gradients, scaling, step size, simultaneous updates	4; 6.1–6.2
RSS, MSE, TSS, R^2	6.3; 7.2
kNN regression and classification	7.1–7.3
Validation, CV, preprocessing, refitting	1.2; 5; 7.2; 7.4(d)
Ridge, lasso, coefficient penalties	1.3; 5; 6.1–6.4
Bias, variance, irreducible noise, learning curves	3; 5
Independent evaluation and Hoeffding for zero–one loss	7.4