Final Review, Day 1 (Wednesday)

EECS 398: Practical Data Science, Winter 2025

practicaldsc.org • github.com/practicaldsc/wn25 • 📣 See latest announcements here on Ed

Agenda 📅

7 → 8 → 9 → 1 → 4 → 5

- We'll work through (some of) the problems in this worksheet: https://study.practicaldsc.org/fi-review-wednesday/index.html
- I'll post these annotated slides after lecture.
- So that we focus our time on more challenging concepts, tell me which questions you want me to cover by voting here:

practicaldsc.org/q

vote

Study/exam-taking tips

- Read the entire exam before writing anything!
  - → Start with Qs you think are easiest
- Practice problems! tells you where to study
  - → Put questions you got wrong on the notes sheet
  - → Replicate the exam environment as best as possible
    - → Time yourself!
- Active recall
  - → Try and teach the concepts to others (do this in groups)

Problem 1

$\vec{u}, \vec{v} \in \mathbb{R}^3$

Consider the vectors $\vec{u}$ and $\vec{v}$, defined below.

$$\vec{u} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \qquad \vec{v} = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}$$

We define $X \in \mathbb{R}^{3 \times 2}$ to be the matrix whose first column is $\vec{u}$ and whose second column is $\vec{v}$.

$$X = \begin{bmatrix} | & | \\ \vec{u} & \vec{v} \\ | & | \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{bmatrix}$$

Problem 1.1

In this part only, let $\vec{y} = \begin{bmatrix} -1 \\ k \\ 252 \end{bmatrix}$.

$a = -1, \; b = 252$

Find a scalar k such that $\vec{y}$ is in $\text{span}(\vec{u}, \vec{v})$. Give your answer as a constant with no variables.

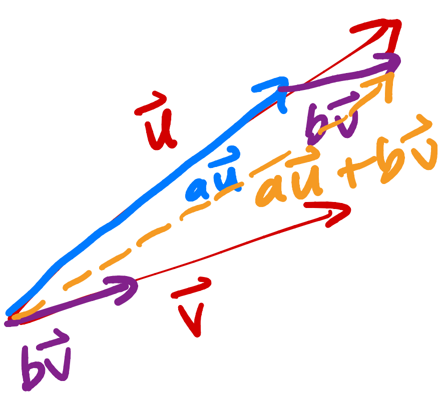
$$a\underset{\vec{u}}{\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}} + b\underset{\vec{v}}{\begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}} = \begin{bmatrix} a \\ b \\ b \end{bmatrix} = \begin{bmatrix} -1 \\ k=252 \\ 252 \end{bmatrix} \Rightarrow b = 252 \Rightarrow \boxed{k = 252}$$

$\vec{u} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$ $\vec{v} = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}$

a linear combination of $\vec{u}$, $\vec{v}$ is any vector that can be written as

$$a\vec{u} + b\vec{v}\ , \quad a, b \in \mathbb{R}$$

span $(\vec{u}, \vec{v})$ = set of all possible linear combinations



Problem 1

$$\vec{w}^* = (X^TX)^{-1}X^T\vec{y}$$

(if X^TX is invertible)

Consider the vectors $\vec{u}$ and $\vec{v}$, defined below.

$$\vec{u} = \begin{bmatrix}1\\0\\0\end{bmatrix} \qquad \vec{v} = \begin{bmatrix}0\\1\\1\end{bmatrix}$$

We define $X \in \mathbb{R}^{3\times 2}$ to be the matrix whose first column is $\vec{u}$ and whose second column is $\vec{v}$.

Problem 1.2

$$X^TX = \underbrace{\begin{bmatrix}1&0&0\\0&1&1\end{bmatrix}}_{X^T}^{2\times 3}\underbrace{\begin{bmatrix}1&0\\0&1\\0&1\end{bmatrix}}_{X}^{3\times 2} = \begin{bmatrix}1&0\\0&2\end{bmatrix}^{2\times 2}$$

Show that:

$$(X^TX)^{-1} = \begin{bmatrix}1&0\\0&\frac{1}{2}\end{bmatrix}$$

$$(X^TX)^{-1}X^T = \begin{bmatrix}1&0&0\\0&\frac{1}{2}&\frac{1}{2}\end{bmatrix}$$

Hint: If $A = \begin{bmatrix}a_1&0\\0&a_2\end{bmatrix}$, *then* $A^{-1} = \begin{bmatrix}\frac{1}{a_1}&0\\0&\frac{1}{a_2}\end{bmatrix}$.

$$(X^TX)^{-1}X^T = \underbrace{\begin{bmatrix}1&0\\0&2\end{bmatrix}}_{(X^TX)^{-1}}\underbrace{\begin{bmatrix}1&0&0\\0&1&1\end{bmatrix}}_{X^T} = \begin{bmatrix}\quad&\quad\end{bmatrix}$$

Problem 1

Consider the vectors $\vec{u}$ and $\vec{v}$, defined below.

$$\vec{u} = \begin{bmatrix}1\\0\\0\end{bmatrix} \qquad \vec{v} = \begin{bmatrix}0\\1\\1\end{bmatrix}$$

We define $X \in \mathbb{R}^{3\times 2}$ to be the matrix whose first column is $\vec{u}$ and whose second column is $\vec{v}$.

Problem 1.3

In parts 3 and 4 only, let $\vec{y} = \begin{bmatrix}4\\2\\8\end{bmatrix}$.

Find scalars a and b such that $a\vec{u} + b\vec{v}$ is the vector in $\text{span}(\vec{u}, \vec{v})$ that is as close to $\vec{y}$ as possible. Give your answers as constants with no variables.

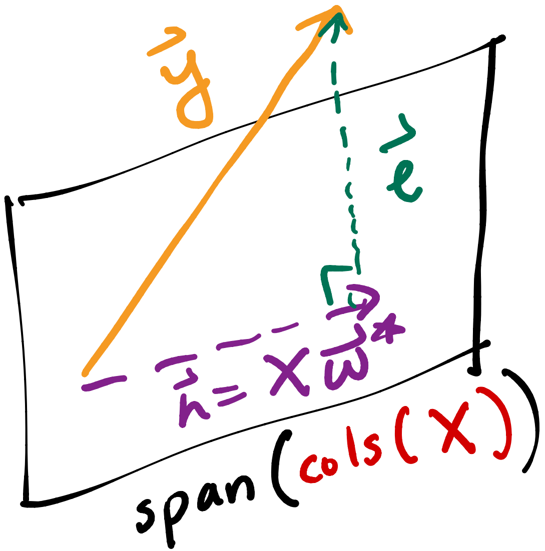
$a=4,\ b=5$

normal equations:

$$\begin{bmatrix}a\\b\end{bmatrix} = (X^TX)^{-1}X^T\vec{y}$$

$$= \begin{bmatrix}1 & 0 & 0\\0 & 1/2 & 1/2\end{bmatrix}\begin{bmatrix}4\\2\\8\end{bmatrix}$$

$$= \begin{bmatrix}4\\ \frac{2}{2}+\frac{8}{2}\end{bmatrix} \boxed{= \begin{bmatrix}4\\5\end{bmatrix}}$$



defining property:

$\vec{e}$ orthogonal to span(X)

$$\Rightarrow \quad X^T\vec{e} = 0$$

$$\Rightarrow \quad X^T(\vec{y} - X\vec{w}^*) = 0 \nearrow \text{normal eq'ns}$$

multiple regression

$$\vec{h} = X\vec{w}$$

$$= \begin{bmatrix} | & | & | \\ 1 & \vec{x}^{(1)} & \vec{x}^{(2)} \\ \vdots & | & | \\ 1 & | & | \end{bmatrix} \begin{bmatrix} w_0 \\ w_1 \\ w_2 \end{bmatrix}$$

$$= w_0 \begin{bmatrix} 1 \\ | \\ \vdots \\ 1 \end{bmatrix} + w_1 \begin{bmatrix} | \\ \vec{x}^{(1)} \\ | \end{bmatrix} + w_2 \begin{bmatrix} | \\ \vec{x}^{(2)} \\ | \end{bmatrix}$$

linear combination of columns of X!

$$X^T(\vec{y}-X\vec{w}^*)=0$$

if X^TX is invertible:

$$\vec{w}^* = (X^TX)^{-1}X^T\vec{y}$$

true even if there's no $\begin{bmatrix}1\\1\\\vdots\\1\end{bmatrix}$ in X;

that just guarantees that $\sum e_i = 0$,

but $X^Te=0$ is true

by design

Problem 1

Consider the vectors $\vec{u}$ and $\vec{v}$, defined below.

$$\vec{u} = \begin{bmatrix}1\\0\\0\end{bmatrix} \qquad \vec{v} = \begin{bmatrix}0\\1\\1\end{bmatrix}$$

We define $X \in \mathbb{R}^{3\times 2}$ to be the matrix whose first column is $\vec{u}$ and whose second column is $\vec{v}$.

$a = 4, \ b = 5$

$\vec{h} = 4\vec{u} + 5\vec{v} = \begin{bmatrix}4\\5\\5\end{bmatrix}$

Problem 1.4

Let $\vec{e} = \vec{y} - (a\vec{u} + b\vec{v})$, where a and b are the values you found in part (c).

What is $\|\vec{e}\|$?

○ 0

[✓ ○ $3\sqrt{2}$]

○ $4\sqrt{2}$

○ 6

○ $6\sqrt{2}$

○ $2\sqrt{21}$

$$\vec{e} = \underset{\vec{y}}{\begin{bmatrix}4\\2\\8\end{bmatrix}} - \underset{\vec{h}}{\begin{bmatrix}4\\5\\5\end{bmatrix}} = \begin{bmatrix}0\\-3\\3\end{bmatrix}$$

$$\|\vec{e}\| = \sqrt{0^2 + (-3)^2 + (3)^2} = \sqrt{18} = \boxed{3\sqrt{2}}$$

Problem 1

Consider the vectors $\vec{u}$ and $\vec{v}$, defined below.

$$\vec{u} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \qquad \vec{v} = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}$$

We define $X \in \mathbb{R}^{3 \times 2}$ to be the matrix whose first column is $\vec{u}$ and whose second column is $\vec{v}$.

Problem 1.5

Is it true that, for any vector $\vec{y} \in \mathbb{R}^3$, we can find scalars c and d such that the sum of the entries in the vector $\vec{y} - (c\vec{u} + d\vec{v})$ is 0?

○ ~~Yes, because $\vec{u}$ and $\vec{v}$ are linearly independent.~~

○ ~~Yes, because $\vec{u}$ and $\vec{v}$ are orthogonal.~~

[✓ ○ Yes, but for a reason that isn't listed here.]

○ ~~No, because $\vec{y}$ is not necessarily in $\operatorname{span}(\vec{u}, \vec{v})$.~~

○ ~~No, because neither $\vec{u}$ nor $\vec{v}$ is equal to the vector $\begin{bmatrix} 1 & 1 & 1 \end{bmatrix}^T$.~~

○ ~~No, but for a reason that isn't listed here.~~

True because $\begin{bmatrix} \vdots \end{bmatrix}$ in $\operatorname{span}(\vec{u}, \vec{v})$,

and if $\begin{bmatrix} \vdots \end{bmatrix}$ in $\operatorname{span}(\vec{u}, \vec{v})$,

then $\begin{bmatrix} \vdots \end{bmatrix} \cdot \vec{e} = 0$

$\Rightarrow \sum e_i = 0$

If c, d are chosen optimally, i.e. $\vec{w}^* = \begin{bmatrix} c \\ d \end{bmatrix}$, then by construction,

$$X = \begin{bmatrix} 1 & 0 \\ \vdots & 1 \\ 0 & \vdots \\ 0 & 1 \end{bmatrix}$$

$$X^T \vec{e} = 0$$

$$X^T(\underbrace{\vec{y} - X\vec{w}^*}) = 0 \rightarrow \vec{y} - (c\vec{u} + d\vec{v})$$

$$\begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} | \\ \vec{e} \\ | \end{bmatrix} = 0$$

$$\rightarrow \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} \cdot \vec{e} = 0$$

$$\Rightarrow \sum e_i = 0$$

Homework 8

$$\Rightarrow \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \cdot \vec{e} = 0 \qquad \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} \cdot \vec{e} = 0$$

$$\begin{bmatrix} 1 \\ 1 \end{bmatrix} \cdot \vec{e} = 0$$

from adding the two eq'ns

Problem 1.6

Suppose that $Q \in \mathbb{R}^{100\times 12}$, $\vec{s} \in \mathbb{R}^{100}$, and $\vec{f} \in \mathbb{R}^{12}$. What are the dimensions of the following product?

$$\vec{s}^{T} Q \vec{f}$$

○ [✓ scalar]

○ 12×1 vector

○ 100×1 vector

○ 100×12 matrix

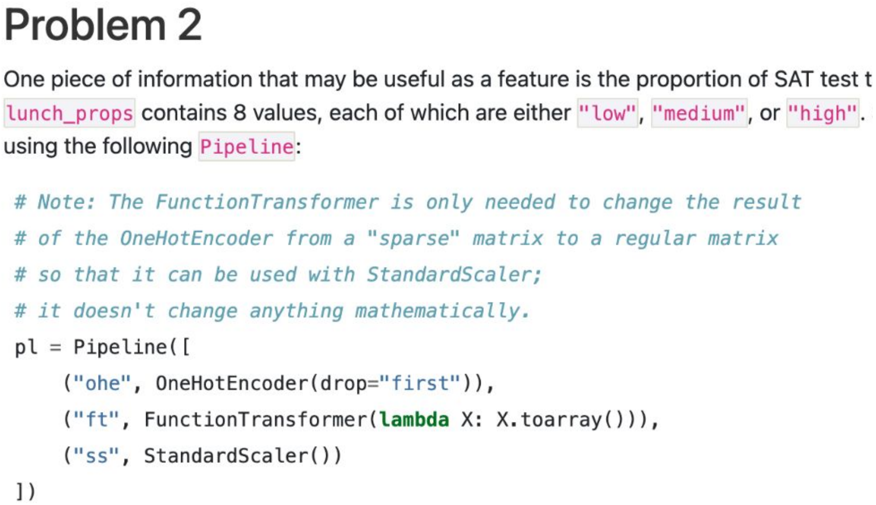
○ 12×12 matrix

○ 12×100 matrix

○ undefined

$$\vec{s}^{\,T}_{\underline{1}\times\cancel{100}}\; Q_{\cancel{100}\times\cancel{12}}\; \vec{f}_{\cancel{12}\times\underline{1}}$$

$$= \text{scalar}_{1\times 1}$$



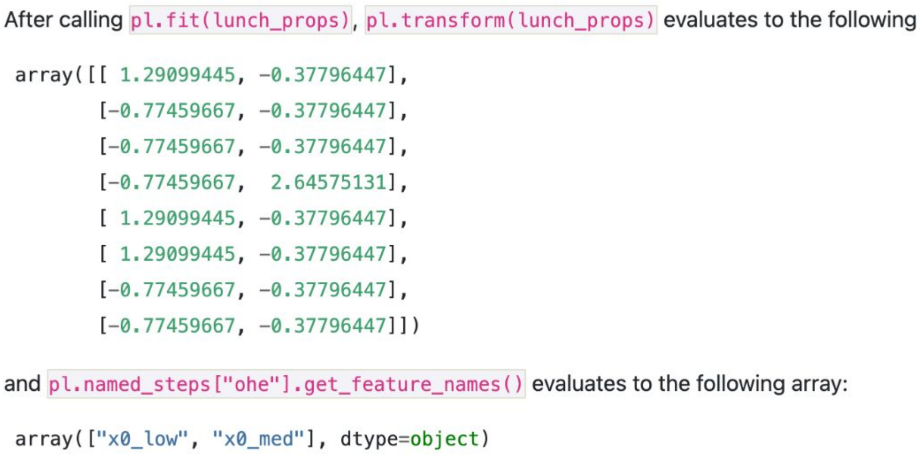
Problem 2

One piece of information that may be useful as a feature is the proportion of SAT test t

`lunch_props` contains 8 values, each of which are either `"low"`, `"medium"`, or `"high"`.

using the following `Pipeline`:

```
# Note: The FunctionTransformer is only needed to change the result
# of the OneHotEncoder from a "sparse" matrix to a regular matrix
# so that it can be used with StandardScaler;
# it doesn't change anything mathematically.
pl = Pipeline([
    ("ohe", OneHotEncoder(drop="first")),
    ("ft", FunctionTransformer(lambda X: X.toarray())),
    ("ss", StandardScaler())
])
```



After calling `pl.fit(lunch_props)`, `pl.transform(lunch_props)` evaluates to the following

```
array([[ 1.29099445, -0.37796447],
       [-0.77459667, -0.37796447],
       [-0.77459667, -0.37796447],
       [-0.77459667,  2.64575131],
       [ 1.29099445, -0.37796447],
       [ 1.29099445, -0.37796447],
       [-0.77459667, -0.37796447],
       [-0.77459667, -0.37796447]])
```

and `pl.named_steps["ohe"].get_feature_names()` evaluates to the following array:

```
array(["x0_low", "x0_med"], dtype=object)
```

Given the above information, we can conclude that `lunch_props` has **(a)** value(s) equal to `"low"`, **(b)** value(s) equal to `"medium"`, and **(c)** value(s) equal to `"high"`. *(Note: Each of (a), (b), and (c) should be positive numbers, such that together, they add to 8.)*

Problem 3

Suppose we have one qualitative variable that that we convert to numerical values using one- hot encoding. We've shown the first four rows of the resulting design matrix below:

a	b	c
1	0	0
1	0	0
0	0	1
0	1	0

Problem 3.1

Say we train a linear model m_1 on these data. Then, we replace all of the 1 values in column **a** with 3's and all of the 1 values in column **b** with 2's and train a new linear model m_2. Neither m_1 nor m_2 have an intercept term. On the training data, the average squared loss for m_1 will be ________ that of m_2.

○ greater than

○ less than

○ equal to

○ impossible to tell

Problem 3

Suppose we have one qualitative variable that that we convert to numerical values using one- hot encoding. We've shown the first four rows of the resulting design matrix below:

a	**b**	**c**
1	0	0
1	0	0
0	0	1
0	1	0

Problem 3.2

To account for the intercept term, we add a column of all ones to our design matrix from part a. That is, the resulting design matrix has four columns: **a** with 3's instead of 1's, **b** with 2's instead of 1's, **c**, and a column of all ones. What is the rank of the new design matrix with these four columns?

○ 1

○ 2

○ 3

○ 4

Problem 4

Suppose we build a binary classifier that uses a song's "track_name" and "artist_names" to predict whether its genre is "Hip-Hop/Rap" (1) or not (0).

For our classifier, we decide to use a brand-new model built into sklearn called the BillyClassifier. A BillyClassifier instance has three hyperparameters that we'd like to tune. Below, we show a dictionary containing the values for each hyperparameter that we'd like to try:

```
hyp_grid = {
  "radius": [0.1, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 100], # 12 total
  "inflection": [-5, -4, -3, -2, -1, 0, 1, 2, 3, 4], # 10 total
  "color": ["red", "yellow", "green", "blue", "purple"] # 5 total
}
```

To find the best combination of hyperparameters for our BillyClassifier, we first conduct a train-test split, which we use to create a training set with 800 rows. We then use GridSearchCV to conduct k-fold cross-validation for each combination of hyperparameters in hyp_grid, with $k = 4$.

$n = 800$

Problem 4.1

When we call GridSearchCV, how many times is a BillyClassifier instance trained in total? Give your answer as an integer.

$$\text{answer} = \left(\text{\# possible hyperparameter combinations}\right) \times k$$

$$= \underbrace{12 \times 10 \times 5}_{600} \times 4$$

$\times 4$

$$= \boxed{2400}$$

1	2	3	4

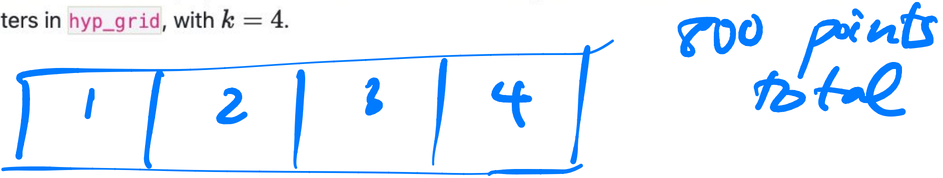
Problem 4

Suppose we build a binary classifier that uses a song's "track_name" and "artist_names" to predict whether its genre is "Hip-Hop/Rap" (1) or not (0).

For our classifier, we decide to use a brand-new model built into sklearn called the BillyClassifier. A BillyClassifier instance has three hyperparameters that we'd like to tune. Below, we show a dictionary containing the values for each hyperparameter that we'd like to try:

```
hyp_grid = {
  "radius": [0.1, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 100], # 12 total
  "inflection": [-5, -4, -3, -2, -1, 0, 1, 2, 3, 4], # 10 total
  "color": ["red", "yellow", "green", "blue", "purple"] # 5 total
}
```

To find the best combination of hyperparameters for our BillyClassifier, we first conduct a train-test split, which we use to create a training set with 800 rows. We then use GridSearchCV to conduct k-fold cross-validation for each combination of hyperparameters in hyp_grid, with $k = 4$.

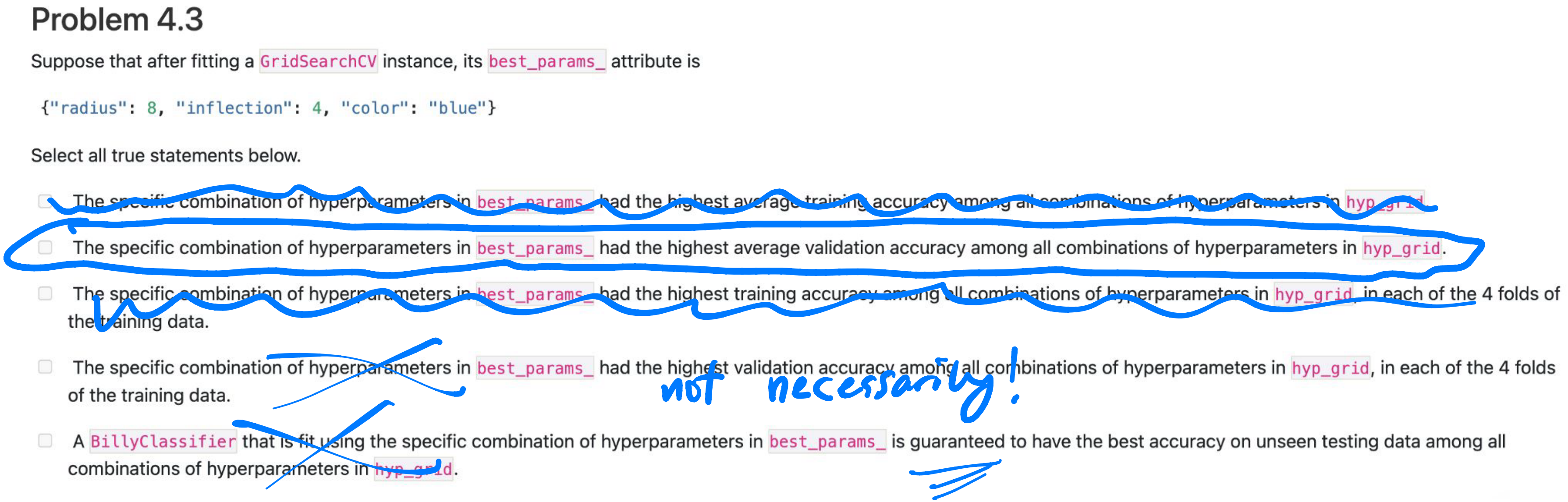


Problem 4.2

In each of the 4 folds of the data, how large is the training set, and how large is the validation set? Give your answers as integers.

size of training set = 600

size of validation set = 200



Problem 4.3

Suppose that after fitting a `GridSearchCV` instance, its `best_params_` attribute is

```
{"radius": 8, "inflection": 4, "color": "blue"}
```

Select all true statements below.

□ ~~The specific combination of hyperparameters in `best_params_` had the highest average training accuracy among all combinations of hyperparameters in `hyp_grid`.~~

[✓ □ The specific combination of hyperparameters in `best_params_` had the highest average validation accuracy among all combinations of hyperparameters in `hyp_grid`.]

□ ~~The specific combination of hyperparameters in `best_params_` had the highest training accuracy among all combinations of hyperparameters in `hyp_grid`, in each of the 4 folds of the training data.~~

□ The specific combination of hyperparameters in `best_params_` had the highest validation accuracy among all combinations of hyperparameters in `hyp_grid`, in each of the 4 folds of the training data.

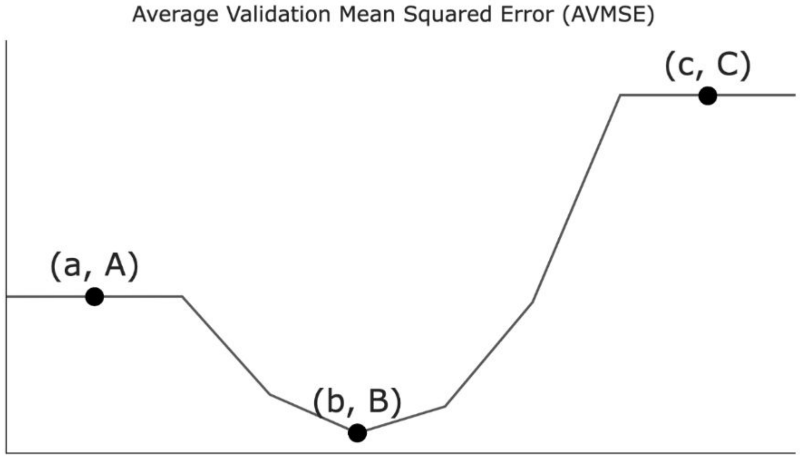
not necessarily!

□ A `BillyClassifier` that is fit using the specific combination of hyperparameters in `best_params_` is guaranteed to have the best accuracy on unseen testing data among all combinations of hyperparameters in `hyp_grid`.

Problem 5

Suppose we want to use LASSO (i.e. minimize mean squared error with L_1 regularization) to fit a linear model that predicts the number of ingredients in a product given its price and rating.

Let λ be a non-negative regularization hyperparameter. Using cross-validation, we determine the average validation mean squared error — which we'll refer to as AVMSE in this question — for several different choices of λ. The results are given below.



Problem 5.1

As λ increases, what happens to model complexity and model variance?

Problem 5.2

What does the value A on the graph above correspond to?

○ The AVMSE of the λ we'd choose to use to train a model.

○ The AVMSE of an unregularized multiple linear regression model.

○ The AVMSE of the constant model.

Problem 6

Suppose we want to minimize the function $R(h) = e^{(h+1)^2}$

Problem 6.1

Without using gradient descent or calculus, what is the value h^* that minimizes $R(h)$?

Problem 6

Suppose we want to minimize the function $R(h) = e^{(h+1)^2}$

Problem 6.2

Now, suppose we want to use gradient descent to minimize $R(h)$. Assume we use an initial guess of $h^{(1)} = 0$. What is $h^{(1)}$? Give your answer in terms of a generic step size, α, and other constants. (e is a constant.)

Problem 6

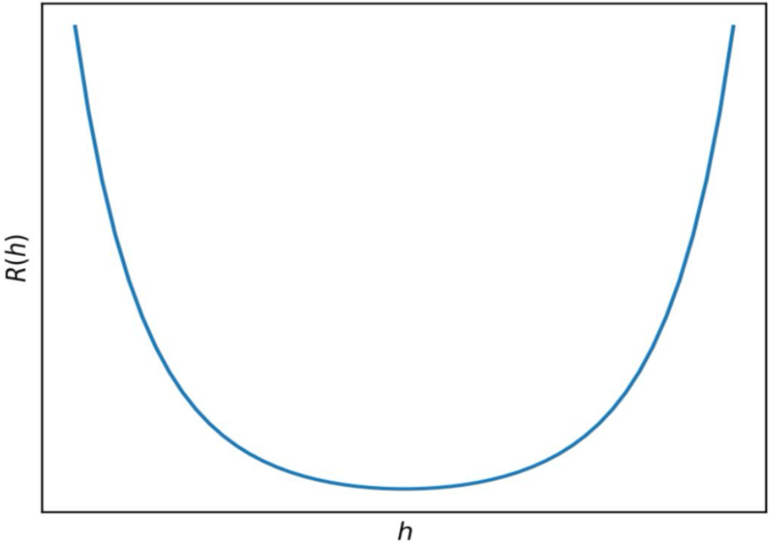
Suppose we want to minimize the function $$R(h) = e^{(h+1)^2}$$

Problem 6.3

Using your answers from the previous two parts, what should we set the value of α to be if we want to ensure that gradient descent finds h^* after just one iteration?

Problem 6.4

Below is a graph of $R(h)$ with no axis labels.



True or False: Given an appropriate choice of step size, α, gradient descent is guaranteed to find the minimizer of $R(h)$.

Problem 7

7.1: constant classifier (more about pandas; not in scope)

We decide to build a classifier that takes in a state's demographic information and predicts whether, in a given year:

- The state's mean math score was greater than its mean verbal score (1), or
- the state's mean math score was less than or equal to its mean verbal score (0).

binary classification

Problem 7.2

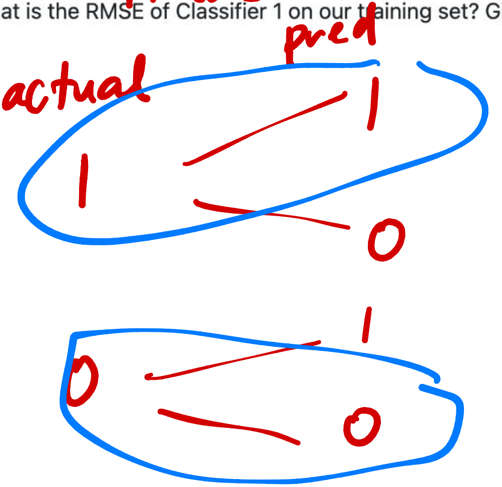
Suppose we train a classifier, named Classifier 1, and it achieves an accuracy of $\frac{5}{9}$ on our training set.

$\frac{5}{9} \cdot n =$ # points classified correctly

Typically, root mean squared error (RMSE) is used as a performance metric for regression models, but mathematically, nothing is stopping us from using it as a performance metric for classification models as well.

$RMSE = \sqrt{MSE}$

What is the RMSE of Classifier 1 on our training set? Give your answer as a **simplified fraction**.



5/9 of the data falls here

$$RMSE = \sqrt{MSE} = \sqrt{\frac{1}{n}\sum(\text{actual}-\text{pred})^2}$$

$$= \sqrt{\frac{1}{\cancel{n}}\left(\frac{5}{9}\cancel{n}\cdot 0^2 + \frac{4}{9}\cancel{n}\cdot 1^2\right)}$$

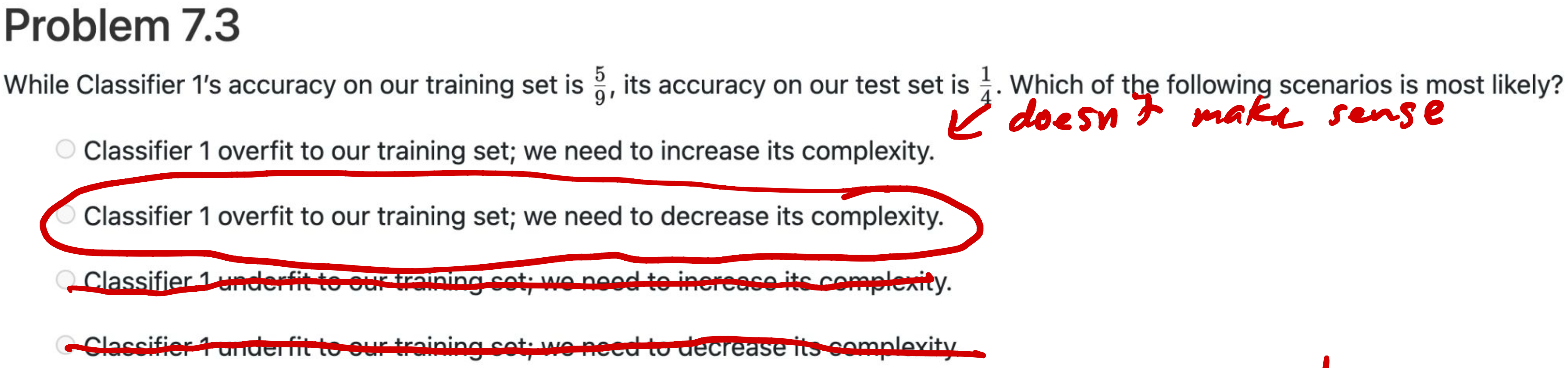
$$= \sqrt{\frac{4}{9}} = \boxed{\frac{2}{3}}$$

Problem 7.2

Suppose we train a classifier, named Classifier 1, and it achieves an accuracy of $\frac{5}{9}$ on our training set.

Typically, root mean squared error (RMSE) is used as a performance metric for regression models, but mathematically, nothing is stopping us from using it as a performance metric for classification models as well.

What is the RMSE of Classifier 1 on our training set? Give your answer as a **simplified fraction**.



Problem 7.3

While Classifier 1's accuracy on our training set is $\frac{5}{9}$, its accuracy on our test set is $\frac{1}{4}$. Which of the following scenarios is most likely?

○ Classifier 1 overfit to our training set; we need to increase its complexity. doesn't make sense

[✓ ○ Classifier 1 overfit to our training set; we need to decrease its complexity.]

○ ~~Classifier 1 underfit to our training set; we need to increase its complexity.~~

○ ~~Classifier 1 underfit to our training set; we need to decrease its complexity.~~

model is too good / overfit to training set ;

make it less complex

For the remainder of this question, suppose we train another classifier, named Classifier 2, again on our training set. Its performance on the training set is described in the confusion matrix below. Note that the columns of the confusion matrix have been separately normalized so that each has a sum of 1.

	Actually 0	Actually 1
Predicted 0	0.9	0.4
Predicted 1	0.1	0.6

	A0	A1
P0	TN	FN
P1	FP	TP

$$0.6 = \frac{TP}{\#A1} = \frac{TP}{TP+FN}$$

Problem 7.4

Suppose `conf` is the DataFrame above. Which of the following evaluates to a Series of length 2 whose only unique value is the number `1`?

○ [✓ `conf.sum(axis=0)`]

○ ~~`conf.sum(axis=1)`~~

Problem 7.5

Fill in the blank: the ___ of Classifier 2 is guaranteed to be 0.6.

○ precision $\frac{TP}{\boxed{TP}+\boxed{FP}}$

○ [✓ recall]

Problem 7.6

Suppose a fraction α of the labels in the training set are actually 1 and the remaining $1-\alpha$ are actually 0. The accuracy of Classifier 2 is 0.65. What is the value of α?

Hint: If you're unsure on how to proceed, here are some guiding questions:

- Suppose the number of y-values that are actually 1 is A and that the number of y-values that are actually 0 is B. In terms of A and B, what is the accuracy of Classifier 2? Remember, you'll need to refer to the numbers in the confusion matrix above.
- What is the relationship between A, B, and α? How does it simplify your calculation for the accuracy in the previous step?

$$0.6\alpha + 0.9(1-\alpha) = 0.65$$

	Actually 0	Actually 1
Predicted 0	0.9	0.4
Predicted 1	0.1	0.6
	B	A

→

0.9B	0.4A
0.1B	0.6A

	A0	A1
P0	TN	FN
P1	FP	TP
	B	A

$$\text{accuracy} = 0.65 = \frac{0.6A + 0.9B}{A+B}$$

$$\alpha = \frac{A}{A+B}$$

$$1-\alpha = \frac{B}{A+B}$$

$$0.65 = 0.6\alpha + 0.9(1-\alpha)$$

$$-0.25 = -0.3\alpha \Rightarrow \alpha = \frac{0.25}{0.30} = \boxed{\frac{5}{6}}$$

Problem 8

Suppose we fit a logistic regression model that predicts whether a product is designed for sensitive skin, given its price, $x^{(1)}$, number of ingredients, $x^{(2)}$, and rating, $x^{(3)}$. After minimizing average cross-entropy loss, the optimal parameter vector is as follows:

$$\vec{w}^* = \begin{bmatrix} \boxed{-1} \\ 1/5 \\ -3/5 \\ 0 \end{bmatrix}$$

price

ingredients

rating

In other words, the intercept term is -1, the coefficient on price is $\frac{1}{5}$, the coefficient on the number of ingredients is $-\frac{3}{5}$, and the coefficient on rating is 0.

Consider the following four products:

- **Wolfcare**: Costs \$15, made of 20 ingredients, 4.5 rating $\boxed{\text{no}}$
- **Go Blue Glow**: Costs \$25, made of 5 ingredients, 4.9 rating
- **DataSPF**: Costs \$50, made of 15 ingredients, 3.6 rating
- **Maize Mist**: Free, made of 1 ingredient, 5.0 rating

e.g.

$$\underbrace{\begin{bmatrix} -1 \\ 1/5 \\ -3/5 \\ 0 \end{bmatrix}}_{\vec{w}^*} \cdot \underbrace{\begin{bmatrix} 1 \\ 15 \\ 20 \\ 4.5 \end{bmatrix}}_{\text{Aug}(\vec{x}_{\text{wolfcare}})} = (-1) + 3 - 12 + 0 = -10 < 0$$

$$-10 < 0 \Rightarrow \sigma(-10) < \frac{1}{2} \Rightarrow \boxed{\text{No}}$$

Which of the following products have a predicted probability of being designed for sensitive skin of **at least 0.5 (50%)**? For each product, select Yes or No.

For each product, compute $\vec{w}^* \cdot \text{Aug}(\vec{x}_{\text{product}})$

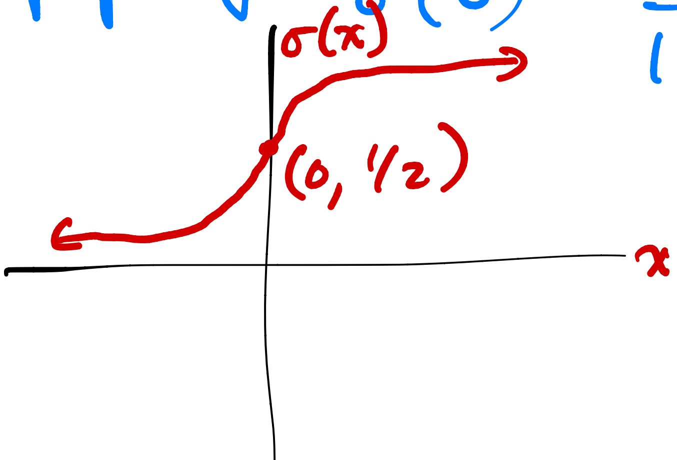
$\Rightarrow$ answer is Yes if $dp \geq 0$, otherwise No

Logistic regression predicts probabilities!

$$P(y_i=1 \mid \vec{x}_i) = \sigma\Big(\underbrace{\vec{w}^* \cdot \text{Aug}(\vec{x}_i)}_{w_0^* + w_1^* x_i^{(1)} + w_2^* x_i^{(2)} + \cdots}\Big)$$

$$\sigma(x) = \frac{1}{1+e^{-x}}, \quad 0 < \sigma(x) < 1$$

Key property: $\sigma(0) = \dfrac{1}{1+e^{0}} = \boxed{\dfrac{1}{2}}$



Problem 9

Consider the following plot of data in $d = 2$ dimensions. We'd like to use k-means clustering to cluster the data into $k = 3$ clusters.

Suppose the crosses represent initial centroids, which are not themselves data points.

Problem 9.1

Which of the following facts are true about the cluster assignment during the first iteration, as determined by these initial centroids?

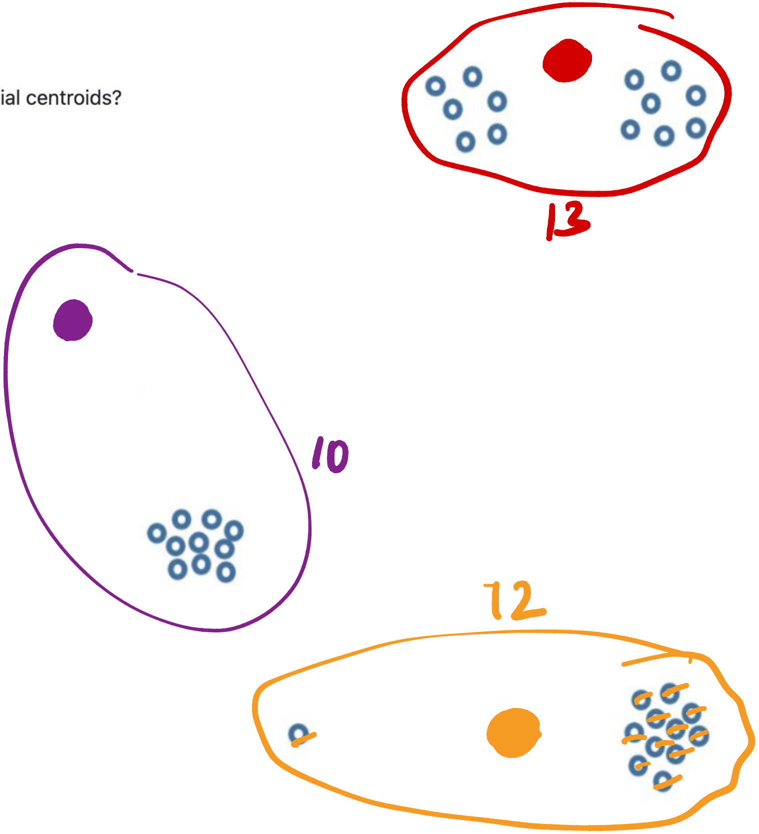
- □ ~~Exactly one cluster contains 11 data~~ points.
- □ ~~Exactly two clusters contain 11 data~~ points.
- □ ~~Exactly one cluster contains at least~~ 12 data points.
- □ [✓ Exactly two clusters contain at least 12 data points.]
- □ None of the above.

10 / $\boxed{12 / 13}$

2 steps that we repeat:

1) color points

2) move centroids



Problem 9

Consider the following plot of data in $d = 2$ dimensions. We'd like to use k-means clustering to cluster the data into $k = 3$ clusters.

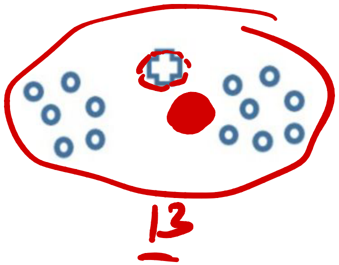
Suppose the crosses represent initial centroids, which are not themselves data points.

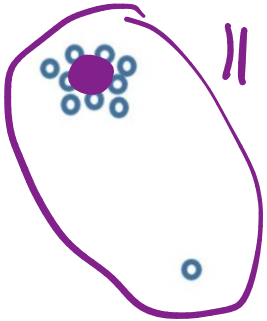
Problem 9.2

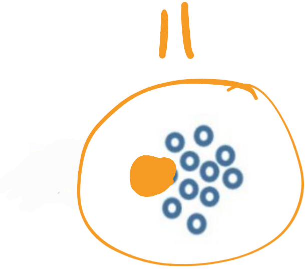
The cross shapes in the plot above represent positions of the initial centroids before the first iteration. Now the algorithm is run for one iteration, after which the centroids have been adjusted.

We are now starting the second iteration. Which of the following facts are true about the cluster assignment during the **second** iteration? Select all that apply.

- □ ~~Exactly one cluster contains 11 data points.~~
- □ [✓ Exactly two clusters contain 11 data points.]
- □ [✓ Exactly one cluster contains at least 12 data points.]
- □ ~~Exactly two clusters contain at least 12 data points.~~
- □ None of the above.







Problem 9

Consider the following plot of data in $d = 2$ dimensions. We'd like to use k-means clustering to cluster the data into $k = 3$ clusters.

Suppose the crosses represent initial centroids, which are not themselves data points.

Problem 9.3

Compare the value of inertia after the end of the second iteration to the value of inertia at the end of the first iteration. Which of the following facts are true? Select all that apply.

- □ The inertia at the end of the second iteration is lower.
- □ ~~The inertia doesn't decrease since there are actually 4 clusters in the data but we are using k-means with $k = 3$.~~
- □ ~~The inertia doesn't decrease since there are actually 5 clusters in the data but we are using k-means with $k = 3$.~~
- □ ~~The inertia doesn't decrease since there is an outlier that does not belong to any cluster.~~
- □ The inertia at the end of the second iteration is the same as at the end of the first iteration.

would be true if terminated after 1 iteration

inertia = sum of the squared distances between each point and its closest centroid

algorithm terminates after second iteration