Final Exam - EECS 398-003, Fall 2024

Full Name:

Uniqname:

UMID:

Room: ◯ DOW 1005 ◯ DOW 1013 ◯ DOW 1017 ◯ Other

Instructions:

- You have 120 minutes to complete this exam.
- **This exam consists of 16 questions, worth a total of 108 points.**
  All 16 questions count towards your Final Exam score.
  Questions 1-6, labeled **Counts towards midterm redemption**, count toward your Midterm Exam Redemption score.
- Write your uniqname in the top right corner of each page in the space provided.
- Please write **clearly** in the provided answer boxes; we will not grade work that appears elsewhere. Completely fill in bubbles and square boxes; if we cannot tell which option(s) you selected, you may lose points.

  ◯ A bubble means that you should only **select one choice**.

  ☐ A square box means you should **select all that apply**.
- You may refer to two double-sided handwritten notes sheets. Other than that, you may not refer to any other resources or technology during the exam (no phones, watches, or calculators).

You are to abide by the University of Michigan/Engineering Honor Code. To receive a grade, please sign below to signify that you have kept the Honor Code pledge.

I have neither given nor received aid on this exam, nor have I concealed any violations of the Honor Code.

Signature:

Data Overview: Skincare

As it gets colder outside, it's important to make sure we're taking care of our skin! In this exam, we'll work with the DataFrame `skin`, which contains information about various skincare products for sale at Sephora, a popular retailer that sells skincare products.

The first few rows of `skin` are shown below, but `skin` has many more rows than are shown.

	Type	Brand	Name	Price	Rating	Num Ingredients	Sensitive
0	Eye cream	PERRICONE MD	PRE:EMPT SERIES™ Brightening Eye Cream	55	4.2	33	1
1	Cleanser	CLINIQUE	Pep-Start 2-in-1 Exfoliating Cleanser	19	3.1	36	0
2	Eye cream	PETER THOMAS ROTH	FIRMx™ 360 Eye Renewal	75	5.0	42	0
3	Treatment	KIEHL'S SINCE 1851	Clearly Corrective™ Dark Spot Solution	50	4.5	24	1
4	Cleanser	PETER THOMAS ROTH	Irish Moor Mud Purifying Cleanser Gel	38	3.6	23	0

The columns in `skin` are as follows:

- `"Type" (str)`: The type of product. There are six different possible types, three of which are shown above.
- `"Brand" (str)`: The brand of the product. As shown above, brands can have multiple products.
- `"Name" (str)`: The name of product. Assume that product names are unique.
- `"Price" (int)`: The price of the product, in a whole number of dollars.
- `"Rating" (float)`: The rating of the product on `sephora.com`; ranges from `0.0` to `5.0`.
- `"Num Ingredients" (int)`: The number of ingredients in the product.
- `"Sensitive" (int)`: `1` if the product is made for individuals with sensitive skin, and `0` otherwise.

Throughout the exam, assume we have already run all necessary import statements.

uniqname: ____________________

Make sure you have read the Data Overview *and* read all of the questions on the exam before you start writing!

Question 1 (8 pts) Counts towards midterm redemption

An expensive product is one that costs **at least $100**.

a) (3 pts) Fill in the blank below with an expression that evaluates to the **proportion** of products in `skin` that are expensive.

b) (5 pts) Fill in the blanks so that the expression below evaluates to the number of brands that sell **fewer than 5** expensive products.

```
skin.groupby(__(i)__).__(ii)__(__(iii)__)["Brand"].nunique()
```

(i): ◯ "Brand" ◯ "Name" ◯ "Price" ◯ ["Brand", "Price"]

(ii): ◯ agg ◯ count ◯ filter ◯ value_counts

(iii):

Question 2 (2 pts) Counts towards midterm redemption

Fill in each blank with **one word** to complete the sentence below.

The SQL keyword for filtering after grouping is __(i)__, and the SQL keyword for querying is __(ii)__.

(i): (ii):

Question 3 (9 pts) **Counts towards midterm redemption**

Consider the Series `small_prices` and `vc`, both of which are defined below.

```
small_prices = pd.Series([
    36, 36, 18, 100, 18, 36, 1, 1, 1, 36]
])

vc = small_prices.value_counts().sort_values(ascending=False)
```

In each of the parts below, select the value that the provided expression evaluates to. If the expression errors, select "Error".

a) (1.5 pts) `vc.iloc[0]`?

◯ 0 ◯ 1 ◯ 2 ◯ 3 ◯ 4
◯ 18 ◯ 36 ◯ 100 ◯ Error ◯ None of these

b) (1.5 pts) `vc.loc[0]`

◯ 0 ◯ 1 ◯ 2 ◯ 3 ◯ 4
◯ 18 ◯ 36 ◯ 100 ◯ Error ◯ None of these

c) (1.5 pts) `vc.index[0]`

◯ 0 ◯ 1 ◯ 2 ◯ 3 ◯ 4
◯ 18 ◯ 36 ◯ 100 ◯ Error ◯ None of these

d) (1.5 pts) `vc.iloc[1]`

◯ 0 ◯ 1 ◯ 2 ◯ 3 ◯ 4
◯ 18 ◯ 36 ◯ 100 ◯ Error ◯ None of these

e) (1.5 pts) `vc.loc[1]`

◯ 0 ◯ 1 ◯ 2 ◯ 3 ◯ 4
◯ 18 ◯ 36 ◯ 100 ◯ Error ◯ None of these

f) (1.5 pts) `vc.index[1]`

◯ 0 ◯ 1 ◯ 2 ◯ 3 ◯ 4
◯ 18 ◯ 36 ◯ 100 ◯ Error ◯ None of these

Question 4 (6 pts) **Counts towards midterm redemption**

Consider the DataFrames `type_pivot`, `clinique`, `fresh`, and `boscia`, defined below.

```
type_pivot = skin.pivot_table(index="Type",
                              columns="Brand",
                              values="Sensitive",
                              aggfunc=lambda s: s.shape[0] + 1)

clinique = skin[skin["Brand"] == "CLINIQUE"]
fresh = skin[skin["Brand"] == "FRESH"]
boscia = skin[skin["Brand"] == "BOSCIA"]
```

Three columns of `type_pivot` are shown below **in their entirety**.

Brand	CLINIQUE	FRESH	BOSCIA
Type			
Cleanser	6.0	NaN	2.0
Eye cream	4.0	NaN	2.0
Face Mask	3.0	4.0	4.0
Moisturizer	3.0	3.0	NaN
Sun protect	2.0	NaN	NaN

In each of the parts below, give your answer as an **integer**.

a) (3 pts) How many rows are in the following DataFrame?

```
clinique.merge(fresh, on="Type", how="inner")
```

b) (3 pts) How many rows are in the following DataFrame?

```
(clinique.merge(fresh, on="Type", how="outer")
         .merge(boscia, on="Type", how="outer"))
```

Question 5 (5 pts) **Counts towards midterm redemption**

Consider a sample of 60 skincare products. The name of one product from the sample is given below:

"our **drops** cream is the best **drops drops** for eye **drops drops** proven formula..."

The total number of terms in the product name above is unknown, but we know that the term **drops** only appears in the name 5 times.

Suppose the TF-IDF of `drops` in the product name above is $\frac{2}{3}$. Which of the following statements are **NOT possible**, assuming we use a base-2 logarithm? Select all that apply.

- [ ] **All** 60 product names contain the term `drops`, **including** the one above.
- [ ] 14 **other** product names contain the term `drops`, in addition to the one above.
- [ ] None of the 59 **other** product names contain the term `drops`.
- [ ] There are 15 terms in the product name above **in total**.
- [ ] There are 25 terms in the product name above **in total**.

Question 6 (3 pts) Counts towards midterm redemption

Suppose `soup` is a BeautifulSoup object representing the homepage of `wolfskin.com`, a Sephora competitor.

Furthermore, suppose `prods`, defined below, is a list of strings containing the name of every product on the site.

```
prods = [row.get("prod") for row in soup.find_all("row", class_="thing")]
```

Given that `prods[1]` evaluates to `"Cleansifier"`, which of the following options describes the source code of `wolfskin.com`?

- Option 1:

```
<row class="thing">prod: Facial Treatment Essence</row>
<row class="thing">prod: Cleansifier</row>
<row class="thing">prod: Self Tan Dry Oil SPF 50</row>
...
```

- Option 2:

```
<row class="thing" prod="Facial Treatment Essence"></row>
<row class="thing" prod="Cleansifier"></row>
<row class="thing" prod="Self Tan Dry Oil SPF 50"></row>
...
```

- Option 3:

```
<row prod="thing" class="Facial Treatment Essence"></row>
<row prod="thing" class="Cleansifier"></row>
<row prod="thing" class="Self Tan Dry Oil SPF 50"></row>
...
```

- Option 4:

```
<row class="thing">prod="Facial Treatment Essence"</row>
<row class="thing">prod="Cleansifier"</row>
<row class="thing">prod="Self Tan Dry Oil SPF 50"</row>
...
```

◯ Option 1 ◯ Option 2 ◯ Option 3 ◯ Option 4

Question 7 (13 pts)

Consider a dataset of n values, $y_1, y_2, ..., y_n$, all of which are **positive**. We want to fit a constant model, $H(x) = h$, to the data.

Let h_p^* be the optimal constant prediction that minimizes average degree-p loss, $R_p(h)$, defined below.

$$R_p(h) = \frac{1}{n}\sum_{i=1}^{n} |y_i - h|^p$$

For example, h_2^* is the optimal constant prediction that minimizes $R_2(h) = \frac{1}{n}\sum_{i=1}^{n} |y_i - h|^2$.

In each of the parts below, determine the value of the quantity provided. By "the data", we are referring to $y_1, y_2, ..., y_n$.

a) (1.5 pts) h_0^*

- ◯ The standard deviation of the data
- ◯ The variance of the data
- ◯ The mean of the data
- ◯ The median of the data
- ◯ The midrange of the data, $\frac{y_{\min}+y_{\max}}{2}$
- ◯ The mode of the data
- ◯ None of the above

b) (1.5 pts) h_1^*

- ◯ The standard deviation of the data
- ◯ The variance of the data
- ◯ The mean of the data
- ◯ The median of the data
- ◯ The midrange of the data, $\frac{y_{\min}+y_{\max}}{2}$
- ◯ The mode of the data
- ◯ None of the above

c) (1.5 pts) $R_1(h_1^*)$

- ◯ The standard deviation of the data
- ◯ The variance of the data
- ◯ The mean of the data
- ◯ The median of the data
- ◯ The midrange of the data, $\frac{y_{\min}+y_{\max}}{2}$
- ◯ The mode of the data
- ◯ None of the above

d) (1.5 pts) h_2^*

- ◯ The standard deviation of the data
- ◯ The variance of the data
- ◯ The mean of the data
- ◯ The median of the data
- ◯ The midrange of the data, $\frac{y_{\min}+y_{\max}}{2}$
- ◯ The mode of the data
- ◯ None of the above

e) (1.5 pts) $R_2(h_2^*)$

- ◯ The standard deviation of the data
- ◯ The variance of the data
- ◯ The mean of the data
- ◯ The median of the data
- ◯ The midrange of the data, $\frac{y_{\min}+y_{\max}}{2}$
- ◯ The mode of the data
- ◯ None of the above

Now, suppose we want to find the optimal constant prediction, h^*_{U}, using the "Ulta" loss function, defined below.

$$L_U(y_i, h) = y_i(y_i - h)^2$$

f) (1.5 pts) To find h^*_{U}, suppose we minimize average Ulta loss (with no regularization). How does h^*_{U} compare to the mean of the data, M?

◯ $h^*_{\text{U}} > M$ ◯ $h^*_{\text{U}} \geq M$ ◯ $h^*_{\text{U}} = M$ ◯ $h^*_{\text{U}} \leq M$ ◯ $h^*_{\text{U}} < M$

Now, to find the optimal constant prediction, we will instead minimize **regularized** average Ulta loss, $R_\lambda(h)$, where λ is a non-negative regularization hyperparameter:

$$R_\lambda(h) = \left(\frac{1}{n} \sum_{i=1}^{n} y_i(y_i - h)^2 \right) + \lambda h^2$$

It can be shown that $\frac{\partial R_\lambda(h)}{\partial h}$, the derivative of $R_\lambda(h)$ with respect to h, is:

$$\frac{\partial R_\lambda(h)}{\partial h} = -2 \left(\frac{1}{n} \sum_{i=1}^{n} y_i(y_i - h) - \lambda h \right)$$

g) (4 pts) Find h^*, the constant prediction that minimizes $R_\lambda(h)$. Show your work, and put a $\boxed{\text{box}}$ around your final answer, which should be an **expression in terms of y_i, n, and/or λ.**

Question 8 (8 pts)

Suppose we want to fit a simple linear regression model (using squared loss) that predicts the number of ingredients in a product given its price. We're given that:

- The average cost of a product in our dataset is \$40, i.e. $\bar{x} = 40$.
- The average number of ingredients in a product in our dataset is 15, i.e. $\bar{y} = 15$.

The intercept and slope of the regression line are $w_0^* = 11$ and $w_1^* = \frac{1}{10}$, respectively.

a) (3 pts) Suppose Victors' Veil (a skincare product) costs \$40 and has 11 ingredients. What is the squared loss of our model's predicted number of ingredients for Victors' Veil? Give your answer as a **number**.

b) (2 pts) Is it possible to answer part (a) above **just** by knowing $\bar{x}$ and $\bar{y}$, i.e. **without** knowing the values of w_0^* and w_1^*?

○ Yes; the values of w_0^* and w_1^* don't impact the answer to part (a).

○ No; the values of w_0^* and w_1^* are necessary to answer part (a).

c) (3 pts) Suppose x_i represents the price of product i, and suppose u_i represents the **negative price** of product i. In other words, for $i = 1, 2, ..., n$, where n is the number of points in our dataset:

$$u_i = -x_i$$

Suppose U is the design matrix for the simple linear regression model that uses **negative** price to predict number of ingredients. Which of the following matrices could be U^TU?

○ $\begin{bmatrix} -15 & 600 \\ 600 & -30000 \end{bmatrix}$

○ $\begin{bmatrix} 15 & -600 \\ -600 & 30000 \end{bmatrix}$

○ $\begin{bmatrix} -15 & 450 \\ 450 & -30000 \end{bmatrix}$

○ $\begin{bmatrix} 15 & -450 \\ -450 & 30000 \end{bmatrix}$

Question 9 (8 pts)

Suppose we want to fit a multiple linear regression model (using squared loss) that predicts the number of ingredients in a product given its price and various other information.

From the Data Overview page, we know that there are **6** different **types** of products. Assume in this question that there are **20** different product **brands**. Consider the models defined in the table below.

Model Name	Intercept	Price	Type	Brand
Model A	Yes	Yes	No	One hot encoded **without** `drop="first"`
Model B	Yes	Yes	No	No
Model C	Yes	Yes	One hot encoded **without** `drop="first"`	No
Model D	No	Yes	One hot encoded **with** `drop="first"`	One hot encoded **with** `drop="first"`
Model E	No	Yes	One hot encoded **with** `drop="first"`	One hot encoded **without** `drop="first"`

For instance, Model A above includes an intercept term, price as a feature, one hot encodes brand names, and doesn't use `drop="first"` as an argument to `OneHotEncoder` in `sklearn`.

In parts (a) through (c), you are given a model. For each model provided, state the **number** of columns and the rank (i.e. number of linearly independent columns) of the design matrix, X. Some of part (a) is already done for you as an example.

a) (1 pt) **Model A**

Number of columns in X: [ ] Rank of X: [21]

b) (2 pts) **Model B**

Number of columns in X: [ ] Rank of X: [ ]

c) (2 pts) **Model C**

Number of columns in X: [ ] Rank of X: [ ]

d) (3 pts) Which of the following models are **NOT guaranteed** to have residuals that sum to 0?

Hint: Remember, the residuals of a fit model are the differences between actual and predicted y-values, among the data points in the training set.

☐ Model A ☐ Model B ☐ Model C ☐ Model D ☐ Model E

Question 10 (5 pts)

Suppose we want to create polynomial features and use ridge regression (i.e. minimize mean squared error with L_2 regularization) to fit a linear model that predicts the number of ingredients in a product given its price.

To choose the polynomial degree and regularization hyperparameter, we use cross-validation through `GridSearchCV` in `sklearn` using the code below.

```
searcher = GridSearchCV(
    make_pipeline(PolynomialFeatures(include_bias=False),
                  Ridge()),

    param_grid={"polynomialfeatures__degree": np.arange(1, D + 1),
                "ridge__alpha": 2 ** np.arange(1, L + 1)},

    cv=K # K-fold cross-validation.
)
searcher.fit(X_train, y_train)
```

Assume that there are N rows in `X_train`, where N is a multiple of K (the number of folds used for cross-validation).

In each of the parts below, give your answer as an **expression involving** D**,** L**,** K**, and/or** N. Part (a) is done for you as an example.

a) How many combinations of hyperparameters are being considered?

LD

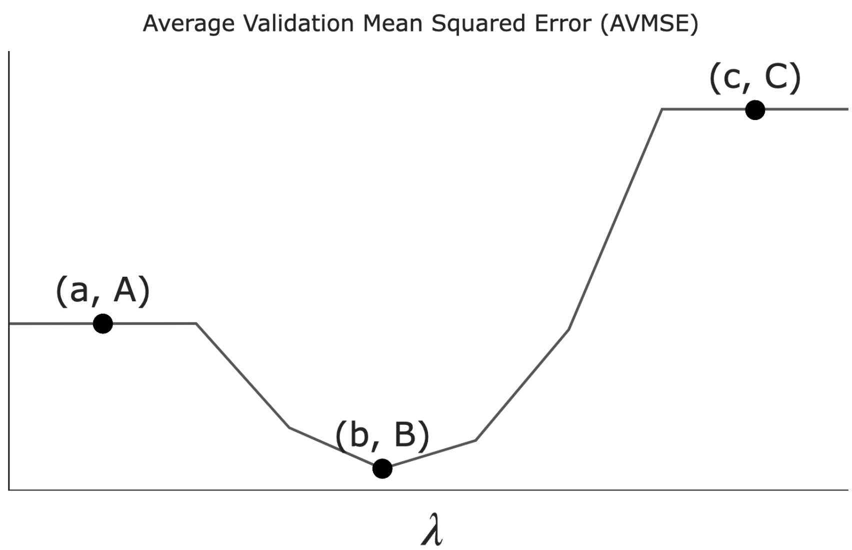
b) (2.5 pts) **Each time** a model is trained, how many points are being used to train the model?

c) (2.5 pts) **In total**, how many times are `X_train.iloc[1]` and `X_train.iloc[-1]` **both** used for training a model **at the same time**? Assume that these two points are in different folds.

Question 11 (8 pts)

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.



a) (2 pts) As λ increases, what happens to model complexity and model variance?

- ◯ Model complexity and model variance both increase.
- ◯ Model complexity increases while model variance decreases.
- ◯ Model complexity decreases while model variance increases.
- ◯ Model complexity and model variance both decrease.

b) (2 pts) 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.

c) (2 pts) What does the value B on the graph above correspond to?

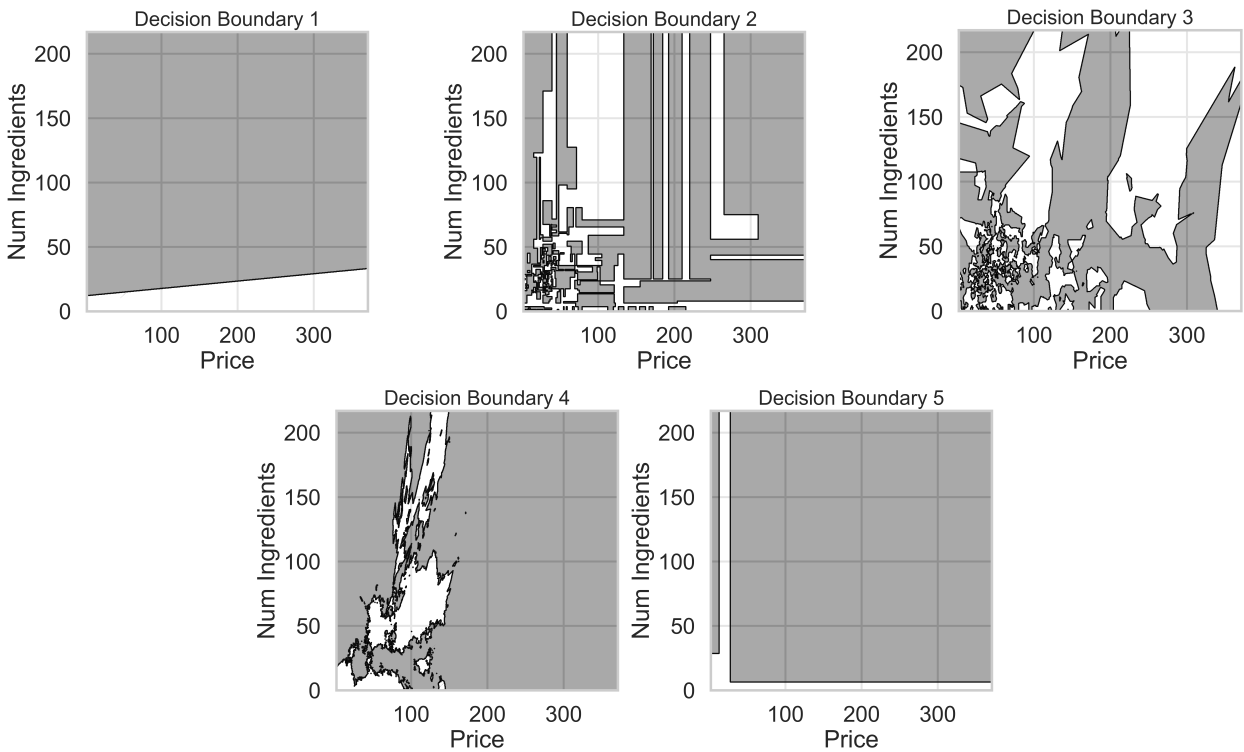
- ◯ The AVMSE of the λ we'd choose to use to train a model.
- ◯ The AVMSE of an unregularized multiple linear regression model.
- ◯ The AVMSE of the constant model.

d) (2 pts) What does the value C 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.

Question 12 (6 pts)

Suppose we fit five different classifiers that predict whether a product is designed for sensitive skin, given its price and number of ingredients. In the five decision boundaries below, the gray-shaded regions represent areas in which the classifier would predict that the product **is** designed for sensitive skin (i.e. predict class 1).



a) (1.5 pts) Which model does Decision Boundary 1 correspond to?

- ◯ k-nearest neighbors with $k = 3$
- ◯ k-nearest neighbors with $k = 100$
- ◯ Decision tree with max depth $= 3$
- ◯ Decision tree with max depth $= 15$
- ◯ Logistic regression

b) (1.5 pts) Which model does Decision Boundary 2 correspond to?

- ◯ k-nearest neighbors with $k = 3$
- ◯ k-nearest neighbors with $k = 100$
- ◯ Decision tree with max depth $= 3$
- ◯ Decision tree with max depth $= 15$
- ◯ Logistic regression

c) (1.5 pts) Which model does Decision Boundary 3 correspond to?

- ◯ k-nearest neighbors with $k = 3$
- ◯ k-nearest neighbors with $k = 100$
- ◯ Decision tree with max depth $= 3$
- ◯ Decision tree with max depth $= 15$
- ◯ Logistic regression

d) (1.5 pts) Which model does Decision Boundary 4 correspond to?

- ◯ k-nearest neighbors with $k = 3$
- ◯ k-nearest neighbors with $k = 100$
- ◯ Decision tree with max depth $= 3$
- ◯ Decision tree with max depth $= 15$
- ◯ Logistic regression

Question 13 (10 pts)

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:

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

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
- **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

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 and justify your answer.

Wolfcare: ◯ Yes ◯ No

Go Blue Glow: ◯ Yes ◯ No

DataSPF: ◯ Yes ◯ No

Maize Mist: ◯ Yes ◯ No

Question 14 (8 pts)

Suppose, again, that we fit a logistic regression model that predicts whether a product is designed for sensitive skin. We're deciding between three thresholds, A, B, and C, all of which are real numbers between 0 and 1 (inclusive). If a product's predicted probability of being designed for sensitive skin is above our chosen threshold, we predict they belong to class 1 (yes); otherwise, we predict class 0 (no).

The confusion matrices of our model on the test set for all three thresholds are shown below.

A

	Pred. 0	Pred. 1
Actually 0	40	5
Actually 1	35	35

B

	Pred. 0	Pred. 1
Actually 0	5	40
Actually 1	10	???

C

	Pred. 0	Pred. 1
Actually 0	10	35
Actually 1	30	40

a) (2 pts) Suppose we choose threshold A, i.e. the **leftmost** confusion matrix. What is the precision of the resulting predictions? Give your answer as an **unsimplified fraction**.

b) (1 pt) What is the missing value (???) in the confusion matrix for threshold B? Give your answer as an **integer**.

c) (3 pts) Using the information in the three confusion matrices, arrange the thresholds from **largest to smallest**. Remember that $0 \leq A, B, C \leq 1$.

◯ $A > B > C$ ◯ $A > C > B$
◯ $B > A > C$ ◯ $B > C > A$
◯ $C > A > B$ ◯ $C > B > A$

d) (2 pts) Remember that in our classification problem, class 1 means the product **is** designed for sensitive skin, and class 0 means the product **is not** designed for sensitive skin. In one or two English sentences, explain which is **worse** in this context and **why**: a false positive or a false negative.

Question 15 (5 pts)

Let $\vec{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$. Consider the function $Q(\vec{x}) = x_1^2 - 2x_1x_2 + 3x_2^2 - 1$.

a) (3 pts) Fill in the blank to complete the definition of $\nabla Q(\vec{x})$, the gradient of Q.

$$\nabla Q(\vec{x}) = \begin{bmatrix} 2(x_1 - x_2) \\ \\ \text{----} \end{bmatrix}$$

What goes in the blank? Show your work, and put a $\boxed{\text{box}}$ your final answer, which should be an **expression involving** x_1 **and/or** x_2.

b) (2 pts) We decide to use gradient descent to minimize Q, using an initial guess of $\vec{x}^{(0)} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ and a learning rate/step size of α.

If after one iteration of gradient descent, we have $\vec{x}^{(1)} = \begin{bmatrix} 1 \\ -4 \end{bmatrix}$, what is α?

○ $\frac{1}{4}$ ○ $\frac{1}{2}$ ○ $\frac{3}{4}$ ○ $\frac{5}{4}$ ○ $\frac{3}{2}$ ○ $\frac{5}{2}$

Question 16 (4 pts)

What is one topic from the second half of the semester that you studied a lot for but wasn't on the exam? **Blank answers will receive no credit!**

Make sure you've written your uniqname in the space provided in the top right corner of every page of this exam!

Congrats on finishing the course — we'll miss you! Feel free to draw us a picture about EECS 398 below :)