Lecture 13: Midterm Review

EECS 398: Practical Data Science, Spring 2025

practicaldsc.org • github.com/practicaldsc/sp25 • 📣 See latest announcements here on Ed

Agenda

- We'll start by working through the first 8 questions of the Fall 2024 Final Exam: study.practicaldsc.org/fa24-final.
- I'll post these annotated slides after lecture.
- There's another review worksheet available at study.practicaldsc.org/mt-review; from 3:30-5PM, I can answer questions about it (or anything else).

	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

Problem 1

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

True
False
False
True
⋮

all work!

Problem 1.1

Write an expression that evaluates to the **proportion** of products in `skin` that are expensive.

① `(skin["Price"] >= 100).mean()`

② `(skin["Price"] >= 100).sum() / skin.shape[0]`

③ `skin[skin["Price"] >= 100].shape[0] / skin.shape[0]`

	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

Problem 1.2

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): (Free response)

DataFrame → one bool

↳ resulting df is not indexed by "Brand";
it has all rows for every "good" brand

filter is when there's a condition on groups (brands)

lambda df: (df["Price"] >= 100).sum() < 5

Boolean Series

Problem 3

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)
```

iloc: "integer location"

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

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

vc / Series

index	
36	4
1	3
18	2
100	1

`vc.iloc[0]`	4
`vc.loc[0]`	error!
`vc.index[0]`	36
`vc.iloc[1]`	3
`vc.loc[1]`	3
`vc.index[1]`	1

vc.index → [36, 1, 18, 100]

Problem 4

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

```
type_pivot = skin.pivot_table(index="Type",
                              columns="Brand",
                              values="Sensitive",
                              aggfunc="count")

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	5	0	1
Eye cream	3	0	1
Face Mask	2	3	3
Moisturizer	2	2	0
Sun protect	1	0	0

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

Problem 4.1

How many rows are in the following DataFrame?

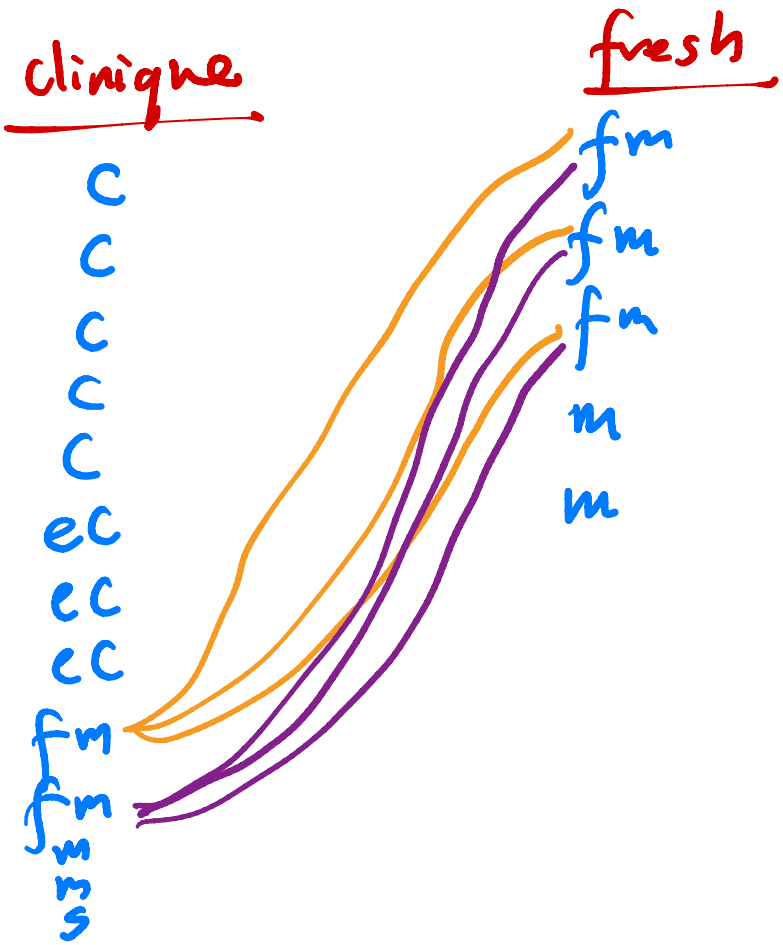
```
clinique.merge(fresh, on="Type", how="inner")
```

Problem 4.2

How many rows are in the following DataFrame?

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

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



3 + 3 + 2 + 2

= 2·3 + 2·2

= 10

Problem 5

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..." $3\ d_1$

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. $n = 60 \rightarrow$ invalid
- [ ] 14 **other** product names contain the term **drops**, in addition to the one above. $n = 15 \rightarrow T = 15$ ✓
- [ ] None of the 59 **other** product names contain the term **drops**. $n = 1 \rightarrow T = T = \frac{3}{2} \cdot 5 \cdot \log_2 60$, not an int
- [ ] There are 15 terms in the product name above **in total**. $T = 15 \rightarrow n = 15$ ✓
- [ ] There are 25 terms in the product name above **in total**. $T = 25$

$$\frac{2}{3} = \frac{1}{5} \cdot \log_2 \frac{60}{n}$$

$$\Rightarrow \frac{10}{3} = \log_2 \frac{60}{n}$$

$$\Rightarrow 2^{\frac{10}{3}} = \frac{60}{n} \Rightarrow n \text{ not whole}$$

constraints

$$\frac{2}{3} = \frac{5}{T} \cdot \log_2 \left(\frac{60}{n}\right)$$

T: len (d_1)
n: # docs with "drops"
both T and n must be integers!

$$\text{TF-IDF}(\text{"drops"}, d_1) = \text{TF}(\text{"drops"}, d_1) \cdot \text{IDF}(\text{"drops"})$$

$$= \frac{\#\text{ of "drops" in } d_1}{\#\text{ words total in } d_1} \times \log_2\left(\frac{\text{number of docs}}{\text{number of docs with "drops"}}\right)$$

$$\frac{2}{3} = \frac{5}{T} \cdot \log_2\left(\frac{60}{n}\right)$$

T: $\text{len}(d_1)$

n: # docs with "drops"

both T and n must be integers!

Problem 6

Suppose `soup` is a BeautifulSoup object representing the homepage of 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 the site?

- 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:

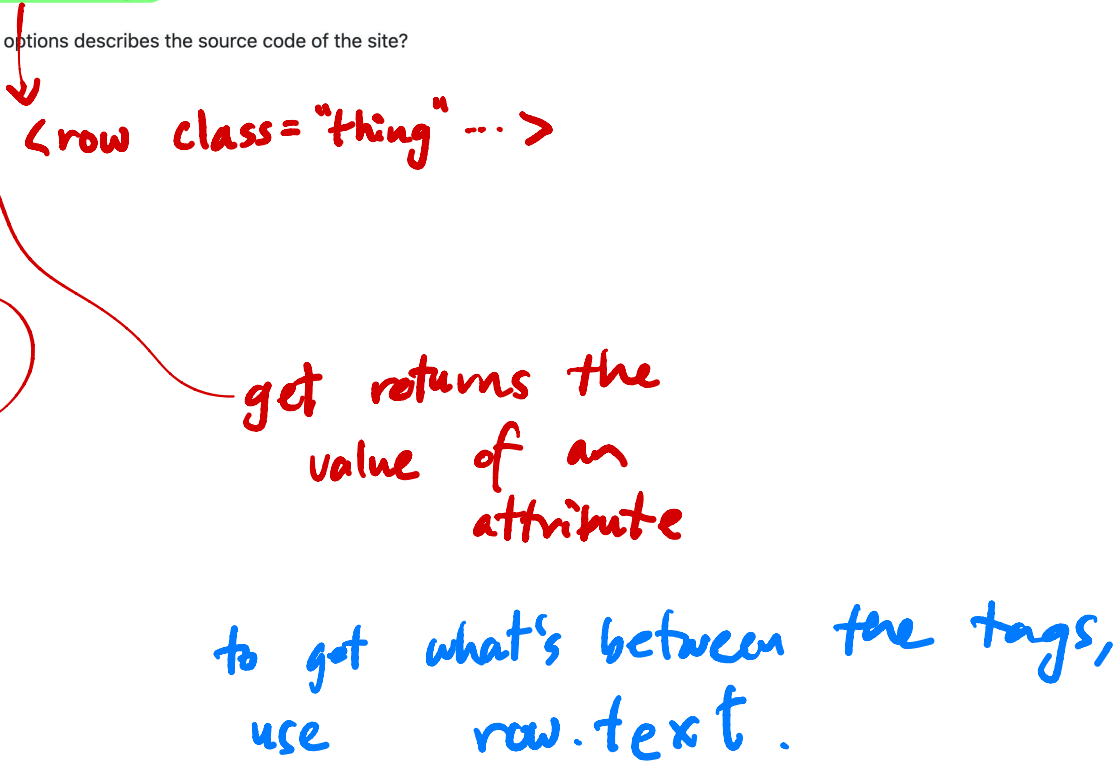
```
<row class="thing" prod="Facial Treatment Essence"></row>
<row class="thing" prod="Cleansifier"></row>
<row class="thing" prod="Self Tan Dry Oil SPF 50"></row>
...
```

- 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>
...
```



Problem 7

Consider a dataset of n values, $y_1, y_2, \ldots, y_n$, 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 $y_1, y_2, \ldots, y_n$.

- 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.

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

$$R_u(h) = \frac{1}{n} \sum_{i=1}^{n} y_i (y_i - h)^2$$

Problem 7.6

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?

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

e.g. $y_1 = 1$ $y_2 = 2$ $y_3 = 100$

$$R_u(h) = \frac{1}{3}\left((1-h)^2 + 2(2-h)^2 + 100(100-h)^2 \right)$$

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:

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

Problem 7.7

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

$$\frac{\partial R}{\partial h} = 0 \rightarrow -2 \left(\frac{1}{n} \sum_{i=1}^{n} y_i (y_i - h) - \lambda h \right) = 0$$

$$\frac{1}{n} \left(\sum_{i=1}^{n} y_i^2 - \sum_{i=1}^{n} y_i h \right) - \lambda h = 0$$

$$\frac{1}{n} \left(\sum_{i=1}^{n} y_i^2 - h \sum_{i=1}^{n} y_i \right) - \lambda h = 0$$

$$\frac{1}{n}\left(\sum_{i=1} y_i^2 - h\sum_{i=1} y_i\right) - \lambda h = 0$$

$$\frac{\sum y_i^2}{n} - h\frac{\sum y_i}{n} - \lambda h = 0$$

$$\frac{\sum y_i^2}{n} = h\left(\frac{\sum y_i}{n} + \lambda\right)$$

$$\frac{\frac{\sum y_i^2}{n}}{\frac{\sum y_i}{n} + \lambda} = h^*$$

$$= \frac{\sum y_i^2}{\sum y_i + n\lambda}$$

multiplied both sides by $\frac{n}{n}$

Problem 8

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.

y

x

Problem 8.1

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**.

$H(x_i) = w_0^* + w_1^* x_i = 11 + \frac{1}{10} x_i$

predict # ingredients given price

predicted # ingredients $= 11 + \frac{1}{10} \cdot 40 = 11 + 4 = 15$

squared loss $= (\text{actual} - \text{predicted})^2$

$= (11 - 15)^2 = 16$

Problem 8

$H(\vec{x}) = w_0^* + w_1^* x_i$

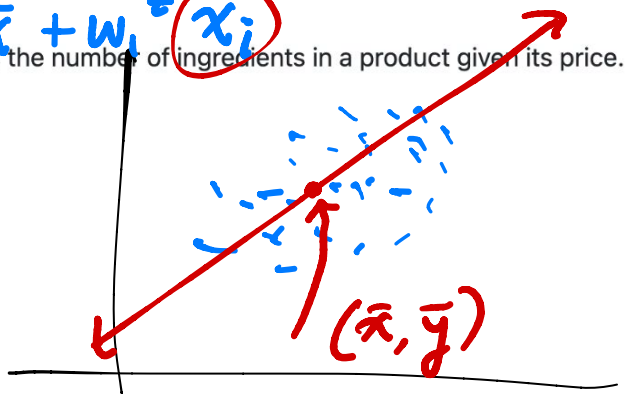
$= \bar{y} - w_1^* \bar{x} + w_1^* x_i$

$= \bar{y}$

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.



Problem 8.2

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).

Fact: If $w_1^* = r \frac{\sigma_y}{\sigma_x} = \cdots$ (the other formulas from Lecture 12)

and $w_0^* = \bar{y} - w_1^* \bar{x}$, then the point $(\bar{x}, \bar{y})$ lies on the regression line!