Kernel Density Estimation

1. We wish to compare the results of kernel density estimation using a gaussian kernel and a boxcar kernel. For $\alpha > 0$, which of the following statements are true? Choose all that apply.

Gaussian Kernel:

$$K_{\alpha}(x, z) = \frac{1}{\sqrt{2\pi\alpha^2}} \exp \left( -\frac{(x - z)^2}{2\alpha^2} \right)$$

Box Car Kernel:

$$B_{\alpha}(x, z) = \begin{cases} \frac{1}{\alpha} & \text{if } -\frac{\alpha}{2} \leq x - z \leq \frac{\alpha}{2} \\ 0 & \text{else} \end{cases}$$

A. Decreasing $\alpha$ for a gaussian kernel decreases the smoothness of the KDE.
B. The gaussian kernel is always better than the boxcar kernel for KDEs.
C. Because the gaussian kernel is smooth, we can safely use large $\alpha$ values for kernel density estimation without worrying about the actual distribution of data.
D. The area under the box car kernel is 1, regardless of the value of $\alpha$.
E. None of the above.

Regular Expressions

2. Which strings contain a match for the following regular expression, "1+1$"? The character "." represents a single space.

- What_is_1+1
- Make_a_wish_at_11:11
- 111_Ways_to_Succeed

3. Given the text:
"<record>ÎFernando ÎPerez Îfperez@berkeley.edu ÎFaculty</record>"
"<record>ÎEdward ÎFang Îedward.fang@berkeley.edu ÎTA</record>"

Which of the following matches exactly to the email addresses (including angle brackets)?
○ <.*@.*>  ○ <^[\^[@]*[^\^]*>>>  ○ <.*\w+\..*>

4. For each pattern specify the starting and ending position of the first match in the string.
The index starts at zero and we are using closed intervals (both endpoints are included).

<table>
<thead>
<tr>
<th>Pattern</th>
<th>abcdefg</th>
<th>abcs!</th>
<th>ab_abc</th>
<th>abc,123</th>
</tr>
</thead>
<tbody>
<tr>
<td>abc*</td>
<td>[0, 2]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[^\s]+</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ab.*c</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[a-z1,9]+</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

5. Write a regular expression that matches strings (including the empty string) that only contain lowercase letters and numbers.

6. Write a regular expression that matches strings that contain exactly 5 vowels.

7. Given that address is a string, use re.sub to replace all vowels with a lowercase letter “o”. For example "123 Orange Street" would be changed to "123 orongo Stroot".

8. Given dates = "October 10, November 11, December 12, January 1", use re.findall to extract all the numbers in the string. The result should look like ["10", "11", "12", "1"].

9. Given the following text in a variable log:
Fill in the regular expression in the variable `pattern` below so that after it executes, day is 26, month is Jan, and year is 2014.

```python
pattern = ...
matches = re.findall(pattern, log)
day, month, year = matches[0]
```

**SQL**

Note: You do not always have to use the JOIN keyword to join sql tables. The following are equivalent:

```sql
SELECT column1, column2
FROM table1, table2
WHERE table1.id = table2.id;

SELECT column1, column2
FROM table1 JOIN table2
ON table1.id = table2.id;
```

10. Describe which records are returned from each type of join.

11. Consider the following real estate schema:

```sql
Homes(home_id int, city text, bedrooms int, bathrooms int, area int)
Transactions(home_id int, buyer_id int, seller_id int,
```
Fill in the blanks in the SQL query to find the id and selling price for each home in Berkeley. If the home has not been sold yet, **the price should be NULL**.

```sql
SELECT ________________
FROM ________________
    JOIN ________________
ON ________________
WHERE ________________;
```