College of Engineering & Applied Sciences

CSPB 3022

Introduction To Data Science With Probability And Statistics

Exam Notes

Taylor Larrechea

2024

Exam 1 Notes

Data Frames

A data frame in Pandas is a two-dimensional, size-mutable, and potentially heterogeneous tabular data structure with labeled axes (rows and columns). It's akin to a spreadsheet or SQL table and is one of the most commonly used Pandas data structures.

You can create a data frame from various sources, such as:

- Lists
- Dictionaries
- NumPy arrays
- CSV files
- SQL databases

The structure of data frames can be summed up by:

- **Rows**: Each row represents a single observation or record.
- **Columns**: Each column represents a variable or feature. Columns can be of different data types (integer, string, float, etc.).
- **Index**: This is the 'key' for rows, similar to an index in a database. It's an immutable array, allowing fast access to data.

Some operations that can be used with data frames are:

- **Data Manipulation**: Adding, deleting, and modifying both rows and columns.
- **Filtering**: Selecting a subset of rows or columns based on some criteria.
- **Sorting and Grouping**: Organizing data based on values in certain columns.
- **Merging and Joining**: Combining multiple data frames.
- **Handling Missing Data**: Identifying and imputing missing values.

Data Frame Operation Examples

Here are some examples of data frame manipulation in pandas:

Creation

```
import pandas as pd

# Creating a data frame from a dictionary
data = {'Name': ['Alice', 'Bob', 'Charlie'],
        'Age': [25, 30, 35],
        'City': ['New York', 'Los Angeles', 'Chicago']}
df = pd.DataFrame(data)

```

Adding A Column

```
# Adding a new column
df['Salary'] = [70000, 80000, 90000]

```

Deleting A Column

```
# Deleting a column
df.drop('Age', axis=1, inplace=True)

```

Filtering Data

```
# Filtering rows where Salary is greater than 75000
high_earners = df[df['Salary'] > 75000]

```

Sorting Data

```
# Sorting data by Salary in descending order
df_sorted = df.sort_values(by='Salary', ascending=False)

```

Merging Data Frames

```
# Creating another data frame
additional_data = pd.DataFrame({'Name': ['Alice', 'Bob'], 'Experience': [5, 10]})

# Merging data frames
merged_df = pd.merge(df, additional_data, on='Name', how='left')

```

Handling Missing Data

```
# Filling missing values with zero
df_filled = df.fillna(0)

```

Reading Data

```
# Reading data from a CSV file
df_from_csv = pd.read_csv('data.csv')

# Writing data to a CSV file
df.to_csv('output.csv', index=False)

```

Combinatorics

Combinatorics is a branch of mathematics dealing with the study of countable, discrete structures and their properties. It's particularly important in computer science, where understanding how to count and arrange objects is crucial for algorithm design, data structure optimization, and problem-solving. Here's a summary of the key concepts in combinatorics:

- **Counting Principles**
  - **The Rule of Sum**: If there are A ways to do something and B ways to do another thing, and these two things cannot happen at the same time, then there are $A + B$ ways to choose one of these actions.
  - **The Rule of Product**: If there are A ways to do something and B ways to do another thing after that, then there are $A \cdot B$ ways to perform both actions.

- **Permutations**
  - Permutations are the arrangements of objects in a specific order.
  - The number of permutations of n distinct objects is $n!$ (n factorial), which is the product of all positive integers up to n.
  - For arranging r objects out of n available objects, the formula is

$$^nP_r = \frac{n!}{(n-r)!}.$$

- **Combinations**
  - Combinations refer to the selection of objects without regard to the order.
  - The number of ways to choose r objects from n different objects is given by

$$\binom{n}{r} = \frac{n!}{r!(n-r)!}.$$

  - Combinations are used when the order doesn't matter.
- **Binomial Theorem**
  - It provides a formula for the expansion of powers of a binomial (sum of two terms).
  - The Binomial Theorem states that:

$$(a+b)^n = \sum_{k=0}^{n} \binom{n}{k} \cdot a^{n-k} \cdot b^k$$

    * This means the expansion is a sum of terms, where the exponents of a start at n and decrease to 0, while the exponents of b start at 0 and increase to n. The coefficients of each term are the corresponding binomial coefficients.
  - The coefficients of the terms in the expansion are the binomial coefficients, which can be calculated using combinations.
- **Binomial Distribution**
  - A binomial distribution is a discrete probability distribution that models the number of successes in a fixed number of independent Bernoulli trials. A Bernoulli trial is an experiment with exactly two possible outcomes, typically termed "success" and "failure".
  - In the context of the binomial distribution, $P(x = k)$ denotes the probability of getting exactly k successes in n trials. The formula for this is

$$P(x = k) = \binom{n}{k} \cdot p^k \cdot (1-p)^{n-k}.$$

    * $\binom{n}{k}$ (read as "n choose k") is the binomial coefficient, representing the number of ways to choose k successes from n trials.
    * p is the probability of success on an individual trial.
    * $1-p$ is the probability of failure (since the trials are binary, the sum of the probabilities of success and failure is 1).
    * p^k is the probability of having k successes.
    * $(1-p)^{n-k}$ is the probability of having $n-k$ failures.