[03-10-2025], Lecture Note: Fairness in Data Selection

[Apoorv Lodhi, Niyati Malik]

CS 516: Responsible Data Science and Algorithmic Fairness; Spring 2025
Abolfazl Asudeh; www.cs.uic.edu/∼ *asudeh/teaching/archive/cs516spring25/*

1 Introduction to Fairness in Database Queries

Automated decision-making systems are widely used in hiring, policing, loans, and more. However, these systems often inherit biases from the data they use, and are increasingly criticized for being discriminatory. Biases in data can stem from selection bias, historical discrimination, and underrepresentation. Fairness in data selection is crucial to mitigate unfair treatment of certain demographic groups.

2 Motivation: Fairness in Range Queries

- Example Scenario: A company aims to identify 'elite' and 'profitable' employees based on salary.
- Issue: If salary being greater than or equal to 65K dollars is used as a threshold, significantly more male employees are selected than female employees.
- Consequence: This reinforces gender disparities in career advancement opportunities.

3 Bias in Data Queries: Problem Statement

Fairness-Aware Range Queries (ICDE 2019 Paper).
Objective:

- Modify database queries to return fairer results while staying similar to the original query.
- Given a biased query result, how can we minimally adjust the query to ensure fairness?

Definition of Fairness: Ensuring demographic groups are equally represented in the query results.
Example: If the original query selected 18% of employees, the modified query ensures gender parity within that selection.

4 Defining Fairness in Range Queries

Group Fairness Concept: Ensures that selected data is proportionally representative of the underlying demographic.

- Formal Definition:
$$|W_r C_r - W_b C_b| \leq \epsilon$$
where:
  - C_r and C_b are the counts of two demographic groups in the query result.
  - W_r and W_b adjust for population sizes.
- Similarity Constraint: Ensure that the new query is as close as possible to the original using the Jaccard similarity measure:
$$\text{SIM}(Q_1, Q_2) = \frac{|\text{out}(D, Q_1) \cap \text{out}(D, Q_2)|}{|\text{out}(D, Q_1) \cup \text{out}(D, Q_2)|}$$

5 Algorithms for Fair Range Queries

5.1 Single-Predicate Range Queries

- Problem Adjusting a single condition (e.g., salary threshold) while maintaining fairness.
- Approach:
  - Compute cumulative sums of demographic groups.
  - Use jump pointers to efficiently find a fair range.

5.2 Jump Pointers Data Structure

- Preprocessing: Store cumulative counts of demographic groups.
- Efficient Query Modification:
  - Instead of adjusting the range step by step, jump directly to the nearest fair boundary.
  - Reduces query modification time from $O(n^2)$ to $O(\log n + \text{disparity})$.

5.3 Multi-Predicate Range Queries

- Problem: Adjusting multiple conditions (e.g., salary & experience level) while maintaining fairness.
- Graph-Based Search:
  - Represent possible range queries as nodes in a graph.
  - Two queries are connected if they differ by one tuple.

5.4 Breadth-First Search (BFSMP)

- Explore neighboring queries to find the closest fair query.
- Guarantees fairness but may be slow.

5.5 Informed Best First Search (IBFSMP)

- Uses heuristics to prioritize promising queries.
- Inspired by the A* search algorithm.
- Finds the most similar fair query faster than BFS.

6 Criticism of the Fair Range Queries Approach

Demographic Parity Definition Issues: Uses absolute count differences rather than ratios (which may be more meaningful). Works well only for binary demographic groups (e.g., Male vs. Female) but not for more complex group structures. Real-World Challenges: Sensitive demographic attributes (e.g., race, gender) are often missing from datasets.

7 Mining Unknown Minority Groups (VLDB 2024)

The Challenge: Missing Sensitive Attributes Many real-world datasets do not include sensitive or demographic attributes such as *race*, *gender*, or *age*. This omission can stem from privacy concerns, legal restrictions, or simply a lack of data collection. However, the absence of these attributes presents a significant challenge for fairness and accountability in machine learning models. Without access to grouping information, practitioners are unable to evaluate whether certain subpopulations are systematically disadvantaged — for example, by experiencing higher error rates or lower representation in training data. This creates an "unknown unknowns" situation: we don't

know what the demographic groups are, and we don't know whether they are being treated fairly by the model.

Illustrative Example (Figure 1 - Toy Dataset): Consider a basketball dataset with features like *height* and *salary*, and a target variable *performance score*. Sensitive attributes like gender are not available. By projecting the data onto a linear direction (a combination of height and salary), we observe a *highly skewed distribution* — a dense cluster (head) and a small group in the tail.

The model performs poorly on this tail group, which is *under-represented* in the data. As discussed in class, this tail corresponds mostly to **female athletes**, revealing a hidden group for which the model underperforms. This shows how high-skew projections can uncover minority groups even without access to explicit sensitive attributes.

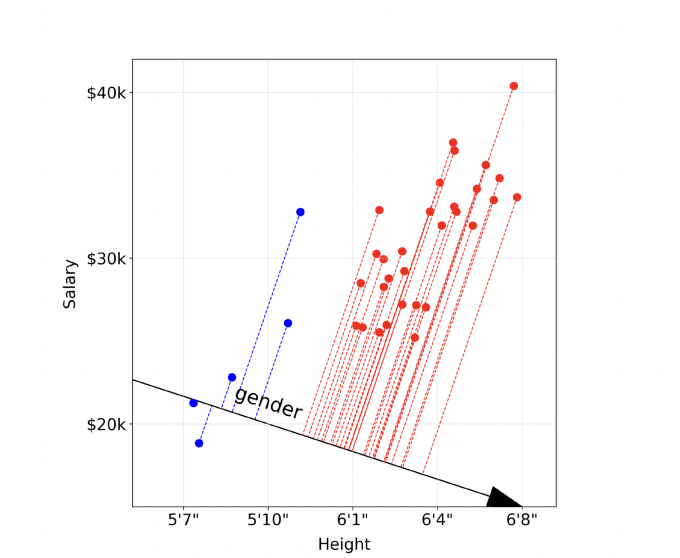

Figure 1: Toy example: High-skew projection in a basketball dataset. The orange region represents the tail containing an under-represented group.

7.1 Formalizing the Minoria Mining Problem

The problem of Minoria Mining focuses on identifying hidden demographic groups in a dataset that are both under-represented and experience poor model performance. These groups are not explicitly labeled and may not correspond to known sensitive attributes such as gender or race.

7.1.1 Problem Setting

The objective is to identify groups in the data that satisfy the following:

- They are **under-represented** — i.e., they appear infrequently in the dataset.
- The model performs **poorly** on them — e.g., high loss or low accuracy.
- The grouping is not predefined — it may be unknown or not explicitly encoded in the dataset.

This creates an "unknown unknowns" scenario, where neither the groups nor their attributes are specified, but they still need to be discovered.

7.1.2 Approach: High-Skew Projections

The key idea is to project the dataset onto different directions in the feature space (i.e., linear combinations of input features) and identify projections with:

- A **high skew** in the distribution of projected values.
- A **tail region** with low sample density, where model performance is significantly worse.

If such a projection is found, and the tail group exhibits higher error than the rest of the dataset, it is flagged as a potential minority group.

7.1.3 Measuring Skew

To quantify skewness in a projection, Pearson's skewness coefficient is used:

$$\text{Skew}(V) = \frac{3(\mu - \nu)}{\sigma}$$

Where:

- μ = mean of the projected values
- ν = median of the projected values
- σ = standard deviation

A higher skew indicates the presence of a long tail, which is used as a signal for possible under-representation.

7.1.4 Projection Computation

Given a data point $x = (x_1, x_2, \ldots, x_d)$ and a direction vector $w = (w_1, w_2, \ldots, w_d)$, the projection onto that direction is calculated as:

$$w^\top x = \sum_{i=1}^{d} w_i x_i$$

7.2 Computational Considerations

Search Space Complexity

- The space of possible projection directions (unit vectors $\mathbf{w}$) is infinite.
- For each direction, projecting n data points takes $O(n)$ time.
- Exhaustively searching over all directions is computationally expensive.

Projection Definition

- A projection of a data point $\mathbf{x}$ onto vector $\mathbf{w}$ is computed as $\mathbf{w}^\top \mathbf{x}$ (e.g., $w_1 x_1 + w_2 x_2$ in 2D).
- The goal is to find the direction $\mathbf{w}$ that results in the maximum skew in the projected data distribution.

Optimization Formulation

- Skew is defined using Pearson's skewness coefficient:

$$\text{Skew} = 3 \times \frac{\text{Mean} - \text{Median}}{\text{Standard Deviation}}$$

- Mean and standard deviation can be computed in $O(n)$ time.
- Median is non-differentiable, making it difficult to use in standard convex or linear optimization frameworks.
- The professor discussed the theoretical possibility of integer programming, though it is neither straightforward nor efficient.

Efficient Algorithms and Target Complexity

- The goal is to find a high-skew projection efficiently, ideally in sub-quadratic time.
- A target time complexity mentioned: $O(n^{4/3})$.
- Achieving this may require techniques from computational geometry or specialized data structures.
- The deterministic "median of medians" algorithm can compute the median in $O(n)$ time.

Motivation and Use Case

- This computational framework supports identifying **unknown**, **underrepresented**, and **underperforming** groups in datasets.
- These groups may not be evident without projection-based analysis, especially in the absence of explicit sensitive attributes.

8 Connecting Both Topics

Fair Range Queries: Modify database queries to ensure fairness in selection-based decisions. Mining Minority Groups: Identify hidden groups that suffer from bias when demographic data is missing. Real-World Connection: If a database query lacks demographic attributes, fairness-aware selection might miss some minority groups. Solution: Combine both approaches—first detect under-represented groups, then adjust queries to ensure fair inclusion.

9 Conclusion

Fairness-aware range queries help reduce bias in database queries. Minority mining is crucial when demographic attributes are missing. Together, these approaches enable responsible AI and data-driven decision-making.