Group X Progress Report:
My Group's Project Name

First Author, Second Author, Third Author
{macid1,macid2,macid3}@mcmaster.ca

1 Introduction

Online dating systems usually optimize for clicks or stated preferences, not mutual compatibility. We study the problem of predicting an individual's ideal partner profile from structured demographic, preference, and ratings data. Given a person A, our model first predicts the set of people A would likely say "yes" to, then composes a representative partner profile B* from the top-k candidates. Next, it predicts who B* would likely say "yes" to, producing a second composite C*. The difference between A and C* captures a preference gap that reveals why A's most desired partners may prefer someone different. This framing supports both recommendation and self-insight, and it generalizes to other matching settings such as mentorship and team formation.

We focus on the Columbia Speed Dating dataset, a tabular dataset with 8,378 rows and 123 columns, and formulate the task as classification for the yes or no decision with ranking objectives layered on top for recommendation quality. Our goals are: (1) build a leakage-free, explainable pipeline that predicts A's decision on B, (2) construct B* and C* to surface preference gaps, and (3) evaluate with Accuracy, F1, ROC-AUC, and Precision@k.

2 Related Work

Our work is informed by several research areas. The first is the original analysis of our dataset, the Columbia Speed Dating Experiment, where Fisman and Iyengar found that attractiveness, fun, and shared interests were highly predictive of matching decisions. Second, our A→B→C loop concept is inspired by two-sided matching theory, most famously represented by the Gale-Shapley algorithm, which underscores that stable matches must account for the preferences of both sides. Third, our approach is related to the broader field of recommender systems, which often use collaborative or content-based filtering to predict preferences, though typically not in a two-hop "preference gap" framework. Fourth, we draw from psychological research on partner preferences, such as the meta-analysis by Finkel, Eastwick, and colleagues, which reviewed the predictive validity of stated ideal partner preferences. Finally, given our goal of providing insights, our work connects to explainable AI (XAI) methods, such as SHAP, which we plan to use to identify which features most influence compatibility predictions.

3 Dataset

We are using the Columbia Speed Dating dataset, as described in our proposal. It consists of **8,378 observations** (individual dates) and **123 columns**.

3.1 Preprocessing and Cleaning

The raw dataset required significant preprocessing. Our pipeline performs the following steps:

- **String Decoding:** Many text columns were encoded as Python byte literals (e.g., 'b'female''). We decoded these into standard UTF-8 strings.
- **Normalization:** All string values were converted to lowercase and stripped of leading/trailing whitespace to ensure consistency (e.g., "Law" and "law" are treated as identical).
- **Missing Values:** We unified various missing value markers (e.g., "na", "n/a", "", "nan") into a single 'pd.NA' representation.
- **Numeric Coercion:** Columns that appeared to be numeric but were stored as objects (e.g., "1.0", "5") were automatically coerced into floating-point types, while preserving categorical ranges (e.g., "[0-1]").

This stable ID generation is crucial for our participant-based train-test split.

4 Features

The model inputs were derived exclusively from pre-event survey responses in the Speed Dating dataset to ensure fairness and avoid post-event leakage. Each training example represents a pairing between two participants (A and B), with features combining demographic, self-assessment, and preference information from both sides.

For participant A, features included demographic attributes (age, gender, race, and field of study), self-ratings (attractive, sincere, intelligence, funny, ambition), personal interests (e.g., sports, music, movies, reading, exercise, hiking, art, shopping), and stated partner preferences (attractive_important, sincere_important, etc.).

For participant B, corresponding partner features were used, including demographic information (age_o, gender_o, race_o), self-ratings (attractive_o, sinsere_o, intelligence_o, funny_o, ambitous_o), and stated partner preferences (pref_o_attractive, pref_o_sincere, etc.). This alignment allowed the model to learn the relationship between what A looks for in a partner and how B describes themselves.

Feature engineering included converting all byte-encoded categorical fields to strings, coercing numeric columns to float representations, and one-hot encoding categorical attributes. We also computed derived attributes such as age difference and same-race indicators to capture potential compatibility signals. No explicit dimensionality reduction or learned embeddings were applied at this stage, as the dataset's moderate size made full feature inclusion tractable.

No additional feature selection was used beyond excluding post-event data (like, match, decision_o) to prevent leakage. Future iterations may explore feature importance rankings from the trained model to prune redundant or weakly predictive features and to better interpret the contribution of personality versus demographic factors.

5 Implementation

We implemented a binary classification model to predict whether a participant (A) would say "yes" to another participant (B) using pre-event survey data from the Speed Dating dataset. The target label is the decision column, where 1 indicates "yes" and 0 indicates "no."

Our baseline model is a simple majority class predictor, which always predicts "no" and achieves approximately 58% accuracy (reflecting the dataset's 58/42 class split). This serves as a lower bound for model performance.

For our primary model, we used a calibrated XGBoost classifier, which is a tree-based gradient boosting algorithm well-suited for mixed numeric and categorical data. The model was trained using log loss (binary cross-entropy) as the objective function, optimized through gradient boosting with decision trees. We applied isotonic calibration on validation data to ensure that predicted probabilities better reflected true likelihoods.

The input features included both A's and B's survey responses: demographic variables (age, gender, race, field), self-ratings, interest scores, and partner preference indicators (*important and pref_o_*). To prevent data leakage, we excluded all post-event information such as like, match, and decision_o.

Training, validation, and testing were split at the participant level rather than by row to avoid information overlap between sets. Optimization used XGBoost's default gradient-based tree boosting with early stopping. Current results yield accuracy ≈ 0.63, F1 ≈ 0.51, and ROC-AUC ≈ 0.66, a notable improvement over the baseline.

Implementation challenges mainly involved cleaning the dataset (many categorical columns were stored as byte strings) and balancing interpretability with model complexity. Our next step is to extend this model to the A $\rightarrow$ B $\rightarrow$ C pipeline, where predictions are used to simulate and analyze multi-stage preference dynamics.

6 Results and Evaluation

The model was evaluated using participant-level data splits to ensure that no individual appeared in multiple subsets, preventing information leakage across training and testing. The dataset was divided into 70% training, 15% validation, and 15% testing sets. This structure allowed for model tuning on the validation set and unbiased performance assessment on unseen participants.

We used a calibrated XGBoost classifier with isotonic calibration applied on the validation data. Calibration was verified using reliability curves and the Brier score to assess the alignment between predicted probabilities and observed outcomes.

Evaluation focused on standard binary classification metrics, including accuracy, precision, recall, F1 score, and ROC-AUC. These metrics collectively measured both discrimination ability and balance between false positives and false negatives. The final model achieved an accuracy of approximately 0.63, F1 score of 0.51, and ROC-AUC of 0.66, representing a meaningful improvement over the 0.58 baseline accuracy from the majority-class predictor.

While we did not employ full cross-validation due to computational constraints, the participant-level split provides a robust approximation of generalization. Future work will consider k-fold validation and top-K ranking metrics (e.g., Precision@K) to better evaluate recommendation-style extensions of the A → B → C pipeline.

7 Feedback and Plans

Write about your plans for the remainder of the project. This should include a discussion of the feedback you received from your TA, and how you plan to improve your approach. Reflect on your implementation and areas for improvement. Refer to item 6. This may be around 0.5 pages.

8 Template Notes

You can remove this section or comment it out, as it only contains instructions for how to use this template. You may use subsections in your document as you find appropriate.

8.1 Tables and figures

See Table 1 for an example of a table and its caption. See Figure 1 for an example of a figure and its caption.

8.2 Citations

Table 1 shows the syntax supported by the style files. We encourage you to use the natbib styles. You can use the command `\citet` (cite in text) to get "author (year)" citations, like this citation to a paper by Gusfield (1997). You can use the command `\citep` (cite in parentheses) to get "(author, year)" citations (Gusfield, 1997). You can use the command `\citealp` (alternative cite without parentheses) to get "author, year" citations, which is useful for using citations within parentheses (e.g. Gusfield, 1997).

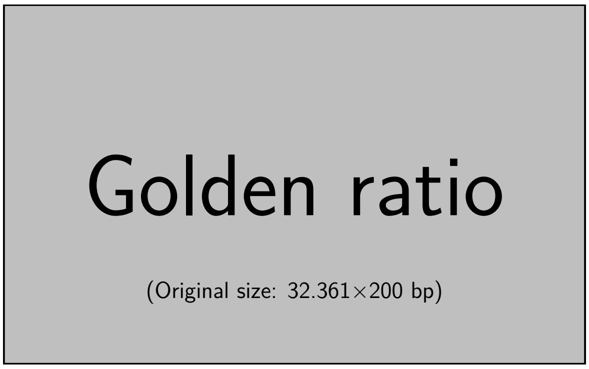

Figure 1: A figure with a caption that runs for more than one line. Example image is usually available through the `mwe` package without even mentioning it in the preamble.

8.3 References

Many websites where you can find academic papers also allow you to export a bib file for citation or bib formatted entry. Copy this into the `custom.bib` and you will be able to cite the paper in the LaTeX. You can remove the example entries.

8.4 Equations

An example equation is shown below:

$$A = \pi r^2 \tag{1}$$

Labels for equation numbers, sections, subsections, figures and tables are all defined with the `\label{label}` command and cross references to them are made with the `\ref{label}` command. This an example cross-reference to Equation 1. You can also write equations inline, like this: $A = \pi r^2$.

Team Contributions

Write in this section a few sentences describing the contributions of each team member. What did each member work on? Refer to item 7.

References

Rie Kubota Ando and Tong Zhang. 2005. A framework for learning predictive structures from multiple tasks and unlabeled data. *Journal of Machine Learning Research*, 6:1817–1853.

Galen Andrew and Jianfeng Gao. 2007. Scalable training of L1-regularized log-linear models. In *Proceedings of the 24th International Conference on Machine Learning*, pages 33–40.

Dan Gusfield. 1997. *Algorithms on Strings, Trees and Sequences*. Cambridge University Press, Cambridge, UK.

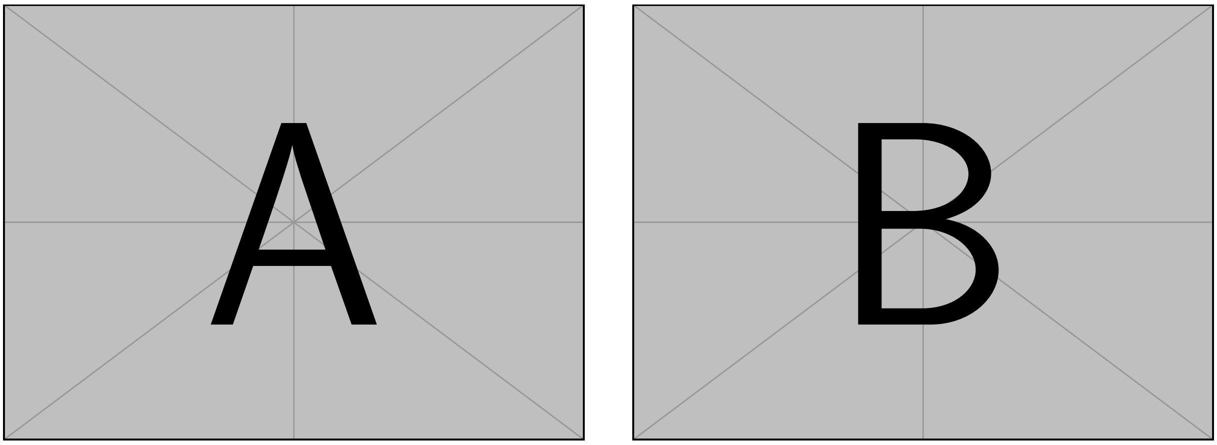

Figure 2: A minimal working example to demonstrate how to place two images side-by-side.

Output	natbib command	ACL only command
(Gusfield, 1997)	`\citep`	
Gusfield, 1997	`\citealp`	
Gusfield (1997)	`\citet`	
(1997)	`\citeyearpar`	
Gusfield's (1997)		`\citeposs`

Table 1: Citation commands supported by the style file.

Mohammad Sadegh Rasooli and Joel R. Tetreault. 2015. Yara parser: A fast and accurate dependency parser. *Computing Research Repository*, arXiv:1503.06733. Version 2.