IDENTIFY FACTORS THAT PREDICT INTRO CS EXPERIENCE BASED ON GENDER

PROJECT OVERVIEW

I once listened to an episode of Star Talk Radio with Neil deGrasse Tyson titled "The Future of Humanity with Elon Musk." Ten minutes into the interview, Musk talks about having sophomoric philosophical wanderings as a student in college. He spent his time musing about the five things that would MOST affect the future of humanity. He thought they were the Internet, sustainable energy, artificial intelligence, rewriting human genetics, and space exploration. Bill Nye, who was also on the show suggested that he would add one more, "the education of women and girls."

A significant change in the future of humanity will be the need to retrain people en masse and getting them into the technical workforce. We are now in a new technological era with autonomous cars driving down our streets and bots like Alexa and Siri becoming an extension of our lives. As automation continues to gain ground, so too are the new industries it helps to create. This new era is creating a new kind of worker, the highly-skilled knowledge worker, in particular, the highly-skilled *technology* knowledge worker.

This shift in the workforce towards highly skilled, technical knowledge workers poses a challenge on the supply side; mostly because of a lack of presence of computer science in K-12 education; the underproduction of post-secondary degrees in computer science; the underrepresentation of women and/or the underrepresentation of ethnic minorities.

I think of this problem as a big-data opportunity where we can kill two birds with one stone. We can invent adult education for workforce readiness en masse while leveraging that opportunity to equalize participation.

As Internet adoption increases, so too will be the opportunity to leverage online education to close the gap between the genders, particularly in emerging countries. A solid understanding of the factors that determine women's participation in computer science can help guide how we design these future learning environments. This project is the start of my journey into understanding those factors.

As part of my doctoral study, I decided to investigate the sociocurricular factors that affect the decision to participate in introductory computer science through a data-driven lens. To do this, I designed a research study examining the role of computer science self-identity centered around the experiences of undergraduates in two introductory computer science classes at UC Berkeley.

PROBLEM STATEMENT

With this project, the problem I am interested in investigating is the gendered experience of the two CS classes in the study. Using machine learning algorithms, I want to identify the leading indicators of the experience of belonging broken down by gender in introductory CS at an elite research university like Berkeley.

To solve this problem, I will undertake the following course of action:

(a) Explore the dataset.
Usually, I would explore the dataset to ensure its integrity and understand the context. But in this case, I will skip this step since I designed the study and collected the data, as such, I am well versed of the context. Further, I have done previous work on this dataset, so I know its boundaries.

(b) Identify features that may be used.
If possible, engineer features that might provide greater discrimination.

(c) With the understanding that this a "classification" task, explore a couple of classifiers that might be well suited for the problem at hand.

(d) Select appropriate classifier based on evaluation metric and tune it for optimality.

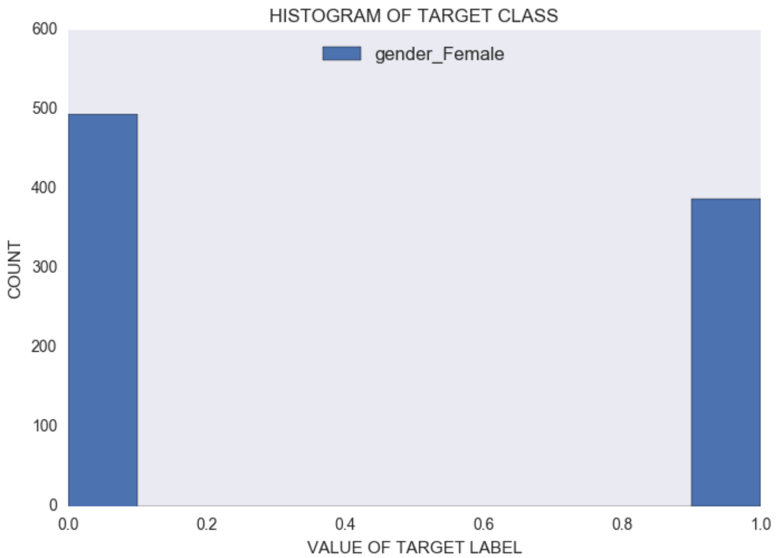
METRICS

Predicting gender in intro CS is a supervised learning problem. To determine the performance of the model, I will be using the F_1 score, i.e., the weighted average of precision and recall as my metric of choice.

I am choosing to use the F_1 score as my metric of evaluation over the accuracy score because my particular dataset has more male students in it than female students. You can see this imbalance in figure 0.1. If I used accuracy, because this imbalance is there, my results could be misleading.

Beyond precision and recall, I will also lean on the result of the confusion matrix for each model. This matrix will let me see which model most accurately *identifies female* students. This will be the **most important** evaluation metric because that is the segment of the student population I am most interested in discovering what determines their experience.

Figure 0.1: **Target Class.** *The histogram shows a slightly unbalanced target dataset with 494 values of {0: male} and 388 values of {1: female}.*



ANALYSIS

DATASET

The dataset used in this project consists of survey responses. A copy of the survey instrument can be found in the appendix of this report. The survey instruments were developed to measure participants' self-reported attitudes along several dimensions:

(a) `atcs`: CS beliefs

(b) `atcsgender`: Gendered belief about CS ability

(c) `atcsjob`: Career driven beliefs about CS

(d) `atct`: Computational thinking beliefs

(e) `blg`: CS belonging

(f) `cltrcmp`: Collegiality

In addition, the survey also collected data around student background:

(a) `prcs`: Prior collegiate CS exposure

(b) `mtr`: CS mentors and role models

(c) University demographics

Majority of the questionnaire uses a 5-point Likert scale (where 1 = Strongly Disagree, 3 = Neutral and 5 = Strongly Agree). A code book was created to facilitate ease of analysis and interpretability of results. The dataset consists of 882 instances with no missing data. Further, there are 494 males and 388 female samples in the dataset.

DATA PREPROCESSING

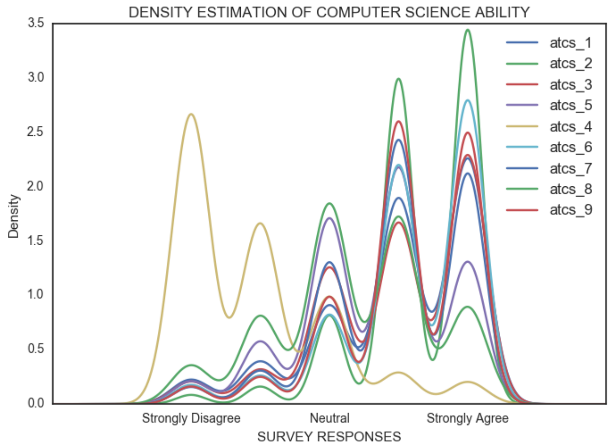
To prepare the data for classification, all features need to be transformed into numeric data. This dataset has several non-numeric columns that need converting. Many of them take on `yes` and `no` values, e.g. `prcs_2`. These can be reasonably convert these into '1'/'0' (binary) values. For the columns whose values are 'Nan', these will be converted to '-1'. Further, whitespaces will be removed from column names with the understanding that the tree plotting algorithm for Xgboost will fail if column names have spaces.

The features were scaled using a minimax scaler to get better output for our SVM. This yielded the following values:

- Strongly Disagree = 0.0
- Disagree = 0.2
- Neutral = 0.6
- Agree = 0.8
- Strongly Agree = 1.0

summarizing the data

Figure 0.2: **Density estimation for dimension atcs.** *Self-reported attitudes about CS.*



I created a density estimation for some dimensions in the data to gain an understanding of the variables and determine if I need to reject some of them, or collapse others. The distributions of most of the dimensions looked very similarly to that of 0.2. Most of the data is either skewed to the left or skewed to the right. As a result, I rejected using descriptive statistics to summarize the data in favor quantiles represented by box plots as can be seen in figure 0.3.

So what does figure 0.3 tell us about the data? From that figure, we can see that the median of this dimension is approximately at the 75 percentile, which based on our Likert scale dataset means most students generally agree with the mostly positive attitudinal questions asked about their CS beliefs. For computational thinking, from figure 0.4 we see that most of the data in this dimension follow a similar distribution.

From 0.5a, I can see that the distribution for the dimension `atcsgender` is really skewed to the right, i.e., most students *strongly disagree* with the statements. That does not come as a surprise, what I found fascinating is that the median for `atcsgender_2` is at the 25 percentile,

Figure 0.3: **Quantiles for dimension** `atcs.` *Self-reported attitudes about CS.*

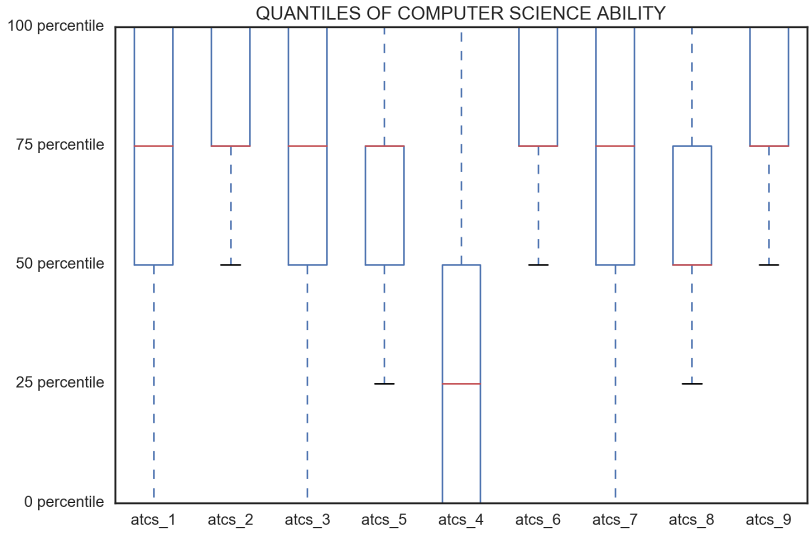
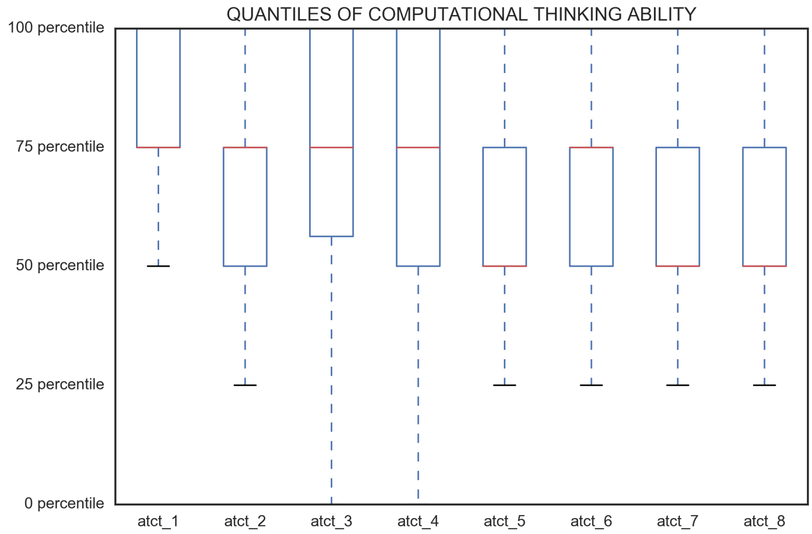


Figure 0.4: **Quantiles for dimension** `atct.` *Self-reported attitudes about computational thinking.*



which corresponds to "neutral." You can see this in the boxplot in figure 0.5b. While students do not agree that women are smarter than men, half of them are undecided about this statement!

- atcsgender_1: Women are less capable of success in CS than men.
- atcsgender_2: Women are smarter than men.
- atcsgender_3: Men have better math and science abilities than women.

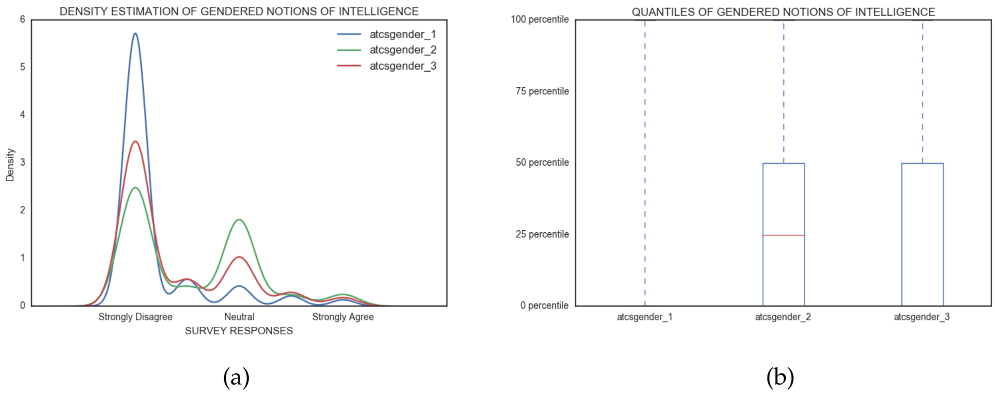


Figure 0.5: **Dimensions of `atcsgender`.** *Figure (a) Density estimation for the dimension. Figure (b) Boxplot for the same dimension.*

ALGORITHMS AND TECHNIQUES

For the problem of determining the factors that predict intro CS experience based on gender, I experimented with four different classifiers, a decision tree classifier, two ensemble methods and a support vector machine:

(a) I selected a Random Forest classifier because it is considered one of the best off-the-shelf learning algorithm, and requires almost no tuning.

(b) I selected an eXtreme Gradient Boosted (XGBoost) trees classifier; which is an advanced implementation of the gradient boosting algorithm. From reading literature on machine learning in practice, the XGBoost classifier has differentiated itself as a classifier that has successfully demonstrated its performance in a wide range of problems. For example, "among the 29 challenge winning solutions published at Kaggle's blog during 2015, 17 solutions used XGBoost."

(c) I selected a Support Vector Machine (SVMs) because they are very robust classifiers and *more importantly*, they have a method to correct for class imbalances.

(d) Finally I selected a Decision Tree classifier because it lends itself to interpretability. For this problem domain, it is not just satisfactory for me to discriminate between male and female students, what I ultimately want is to gain *insights* into what the salient factors around the experience of intro CS are, based on gender.

BENCHMARK

This is novel research, as a result, there are no benchmarks to compare the performance of our classifiers with.

IMPLEMENTATION

I implemented the four learning algorithms. For each of the learners I implemented the baseline algorithm using a stratified shuffle split cross validation with 10 folds and calculated the F_1 scores and looked at the confusion matrices respectively.

Table 0.1: Scores

Result of training the baseline classifiers		
Classifier	Training Score	Prediction Score
SVC	55.16%	54.12%
DecisionTree	49.26%	60.79%
RandomForestClassifier	50.64%	54.64%
XGBClassifier	61.73%	68.37%

From running these baseline classifiers, I selected the xgboost classifier because it had the highest score. In addition, when I looked at the result of the confusion matrix, I decided to use the lowest **false negative** count for the female class as my evaluation metric. From figures 0.6b and 0.6c, I can see that the decision tree and xgboost learners are tied for having the lowest score of false negatives for the female class.

(a) Random Forest

(b) Decision Tree

(c) XgBoost

(d) SVC

Figure 0.6: **Confusion Matrices of Baseline Classifiers**

RESULTS

MODEL EVALUATION AND VALIDATION

I am going to tune my model based on some heuristics about the kinds of value ranges that are suitable for the hyper-parameters I want to learn. I will be using these values ranges for the hyper-parameters:

- Parameters for Tree Booster
  - `max_depth`
    - Maximum depth of tree
    - Range [1, ∞], default 6, tuned on $[4, 6, 8, 10]$
  - `n_estimators`
    - Minimum number of trees
    - Range $[2, \infty]$ default 2, tuned on $range(100, 1100, 100)$
- Task Parameter
  - `learning_rate`
    - Scale the contribution of each tree by learning rate
    - Range $[0, 1]$, tuned on $[0.2222, 0.4444, 0.6666, 0.8888]$

I will implement the tuning using sklearn's `GridSearch` in conjunction with a {k=50 fold} `StratifiedShuffleSplit` function.

RESULTS OF TUNING

Once I performed the search through the hyper-parameter space to find the combination of hyper-parameters that maximized the performance of the selected classifier, I was able to **improve** the previous F_1 score by **6.63%**, to achieve a prediction score of 75%.

From figures 0.7a and 0.7b, one can see that the **false negative** count for the female class has gone from 43 down to 32. This decision cost us a very small increase in the **false positive** count of the male class from 19 to 20. This is not too bad, so I will stick with this improved model.

Looking at figure 0.8, the tuned model has a very complex tree that goes down ten levels, for each of its estimators. This model as segmented the data into 47 distinct types; you can see this by counting the number of leaf nodes.

Here is the final model for classifying gender in introductory CS.

(a) Base Model (b) Tuned Model

Figure 0.7: **Confusion Matrix**

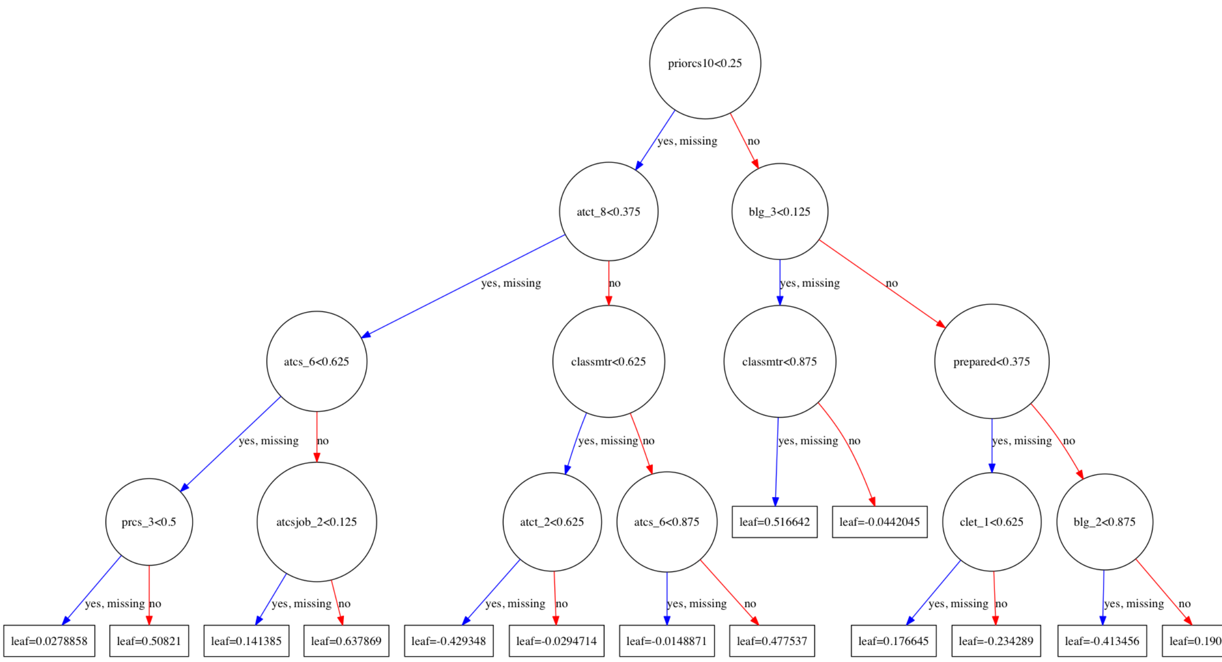


Figure 0.8: **Tuned XgBoost estimator decision tree.**

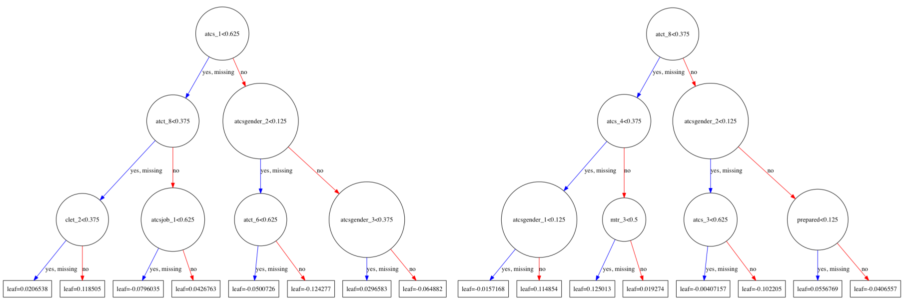
```
Best accuracy obtained: 0.621055664439
Parameters:
    n_estimators: 200
    subsample: 0.7
    learning_rate: 0.2222
    colsample_bytree: 0.6
    max_depth: 10
```

FEATURE IMPORTANCE

There are two things that need consideration when using xgBoost for understanding feature importance: the features that are doing the *most* work in splitting the data, and the automatically generated feature importance ranking that is done in by the xgBoost algorithm.

I plotted some estimators in the xgboost learner to see which features are doing the most work in splitting the data. I chose to focus on the **first** and **second** tree in the ensemble. On simple models, the first two trees may be enough to gain a *strong* understanding. I then compare the output generated by these trees to the features generated by the model's own feature selection algorithm.

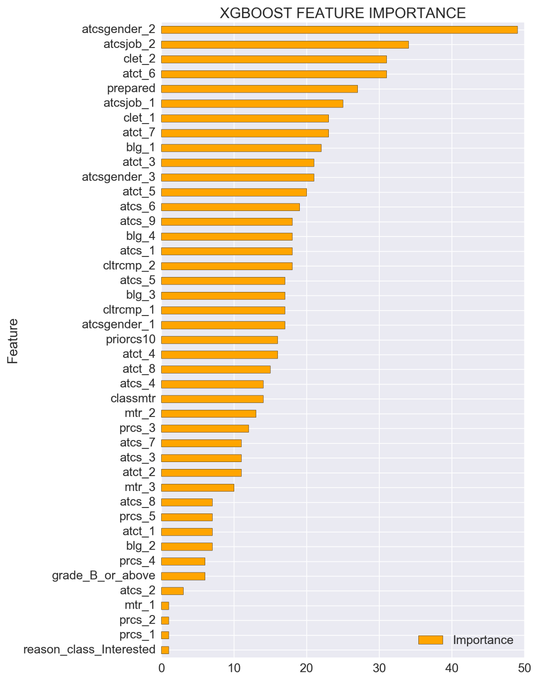
From figures 0.9a and 0.9b, I see that



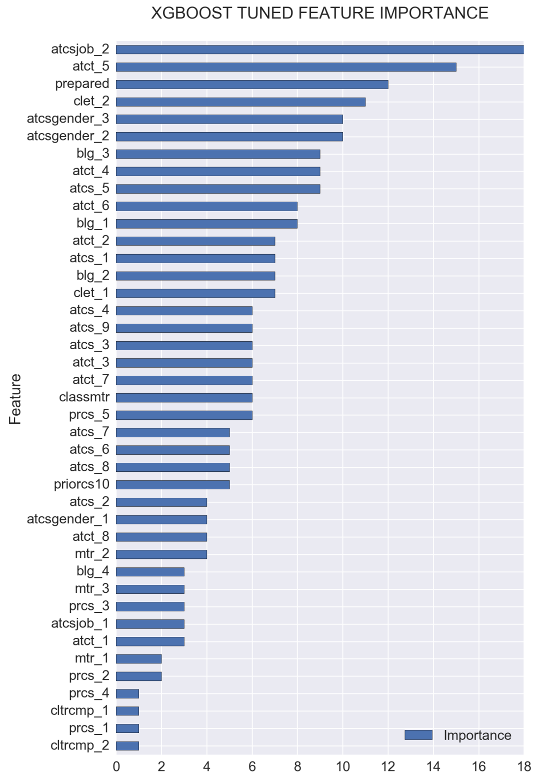
(a) First tree in the ensemble (b) Second tree in the ensemble

Figure 0.9: **XgBoost estimator regression trees**

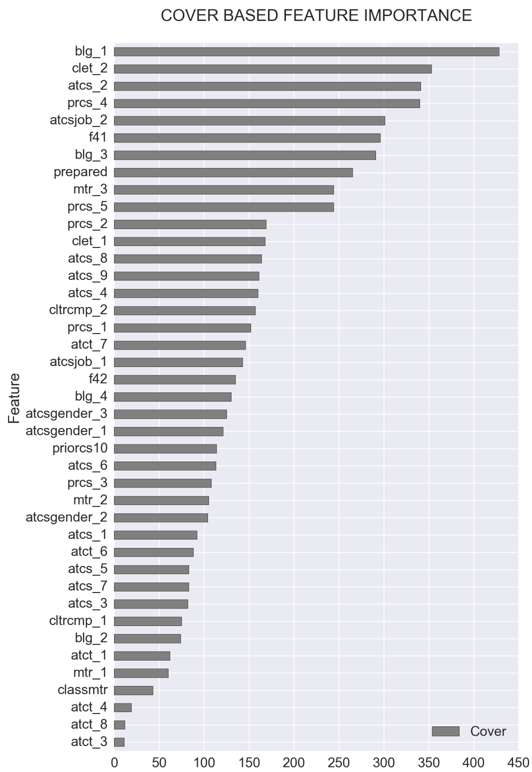
Information gain, lets us know how valuable a feature is in discriminating the dataset. That is, if we know the information gain of a feature, we can know how much it would contribute to the knowledge of the value of the target label.



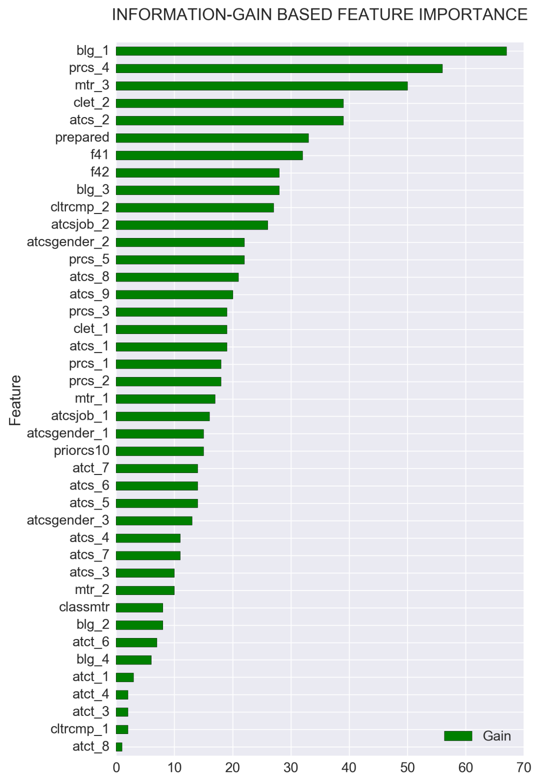
(a) Feature Importance Base Model



(b) Feature Importance Tuned Model



(c) Feature Importance Based on Cover



(d) Feature Importance Based on Information Gain

Figure 0.10: **XgBoost Feature Importance**

Table 0.2: Feature Importance

XGBoost Feature Importance		
Rank	Coded item	Description
1	atcsgender_2	Women are smarter than men.
2	atcsjobs_2	My career goals do not require that I learn computing skills.
3	clet_2	I think about the ethical, legal, and social implications of computing.
4	atct_6	I am good at building things.
5	prepared	How prepared did you feel about this class before it started?
6	atcsjob_1	Knowledge of computing will allow me to secure a good job.
7	clet_1	I work well in teams.
8	atct_7	I am good at ignoring irrelevant details to solve a problem.
9	blg_1	In this class, I feel I belong.
10	atct_3	I am good at using online search tools.

CONCLUSION

REFLECTION

SURVEY INSTRUMENTS

demographics

- Gender [Male, Female, Other]
- What is your reason for taking this class [interested, other]
- What is your major?

attitudes towards computer science

- I like to use Computer Science to solve problems.
- I can learn to understand computing concepts.
- I can achieve good grades (C or better) in computing courses.
- I do not like using computer science to solve problems.
- I am confident that I can solve problems by using computer applications.
- The challenge of solving problems using computer science appeals to me.
- I am comfortable with learning computing concepts.
- I would take additional Computer Science courses if I were given the opportunity.
- I am confident about my abilities with regards to computer science.
- I do think I can learn to understand computing concepts.

career driven beliefs about computer science

- Knowledge of computing will allow me to secure a good job.
- My career goals do not require that I learn computing skills.

attitudes about computational thinking

- I am good at solving a problem by thinking about similar problems I've solved before.
- I have good research skills.
- I am good at using online search tools.

- I am persistent at solving puzzles or logic problems.
- I know how to write computer programs.
- I am good at building things.
- I'm good at ignoring irrelevant details to solve a problem.
- I know how to write a computer program to solve a problem.
- I work well in teams.
- I think about the ethical, legal, and social implications of computing.

Computer Science Mentors and Role Models

- Before I came to UC Berkeley, I knew people who have careers in Computer Science.
- There are people with careers in Computer Science who look like me.
- I have role models within the Computer Science field that look like me.

Identity and Self Efficacy

- In this class, I feel I belong.
- In this class, I feel awkward and out of place
- In this class, I feel like my ideas count
- In this class, I feel like I matter.
- I am comfortable interacting with peers from different backgrounds than my own (based on race, sexuality, etc.)
- I have good cultural competence, or the ability to interact effectively with people from diverse backgrounds.
- Our class materials (e.g., case studies and projects) were relevant and practical

Gendered Belief about Computer Science Ability

- Women are less capable of success in CS than men
- Men have better math and science abilities than women.
- Women are smarter than men.

PRE-COLLEGIATE CS PREPARATION

- Did you take a CS course in High School?
- Did you have exposure to Computer Science before UC Berkeley?
- Did a family member introduce you to Computer Science?
- Did you have a close family member who is a Computer Scientist or is affiliated with computing industry?
- Did your school offer AP CS?
- How prepared did you feel about this class before it started?
- Will you be taking any more CS classes (if so which ones?)
- (For 61A only) Have you taken CS10, The Beauty and Joy of Computing?