Price Predictor

When is the best time to buy a product?

What machine learning algorithm should we use?

Are there any good methods for ensuring our data requests do not look like a DDOS attacks?

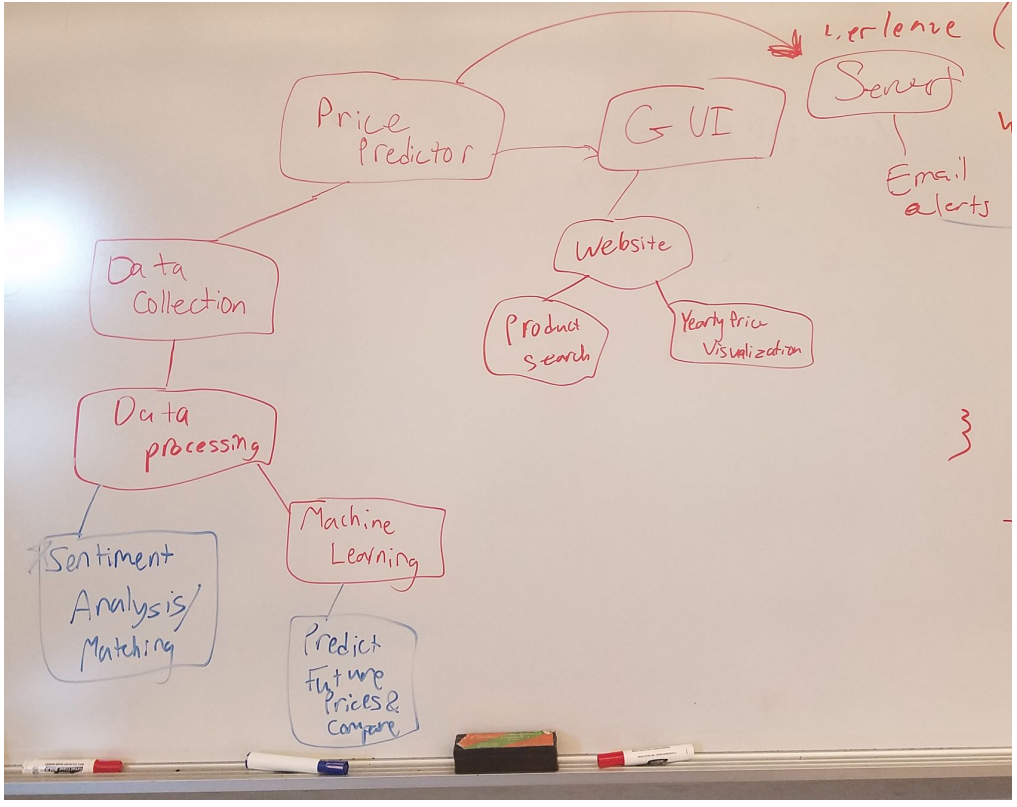
Agenda

- Present our problems
- Brief overviews of potential solution algorithms
- Give sticky notes and ask peers to write down thoughts
- Have an mid presentation discussion with our peers about what machine learning algorithm appears best for our project
- Present our data collection technique
- Have discussion about DDOS attacks

What our project is:

- Scraping Amazon data for product price history
- Using sentiment analysis of reviews to help predict future prices
- Using machine learning for predicting the future prices
- Webapp where users search items and output when the best time to buy a product is
- Website that sends email when the best time to buy a product is

Code Organization and Concept Map



Why we need you :

- We want help to decide which machine learning algorithms are best for our needs
- We want help ensuring our page requests do not look like DDOS attacks

Machine Learning Algorithm: Linear Regression

- When an independent variable is used to predict the value of a dependent variable
  - $Y = a + bx$
- Best at: Baseline predictions, economic predictions, modelling marketing responses
- Pros: fast to train, simple to understand, seldom overfits, easy to train on big data
- Cons: can suffer from outliers, hard to make it fit nonlinear functions

Machine Learning Algorithm: Least squares Regression

- Works like linear regression model, but minimizes the sum of the square of the differences between the observed and the predicted
- Advantages: works best if the trend is linear, maximum likelihood solution
- Disadvantages: unstable, highly influenced by outliers

Machine Learning Algorithm: Ridge Regression

- Ridge regression is like least squares but shrinks the estimated coefficients towards zero.
- Bias is introduced:
  - Bias increases as amount of shrinkage (lamda) increases
  - Variances decreases as amount of shrinkage (lamda) decreases
- Works well when subset of true coefficients are small

$$\sum_{i=1}^{n}\left(y_i-\beta_0-\sum_{j=1}^{p}\beta_j x_{ij}\right)^2+\lambda\sum_{j=1}^{p}\beta_j^2=\mathrm{RSS}+\lambda\sum_{j=1}^{p}\beta_j^2,$$

Machine Learning Algorithm: LASSO

- Least Absolute Shrinkage and Selection Operator
- Regression method that involves penalizing the absolute size of the regression coefficients
- Penalizing allows further estimates to go to zero
- Convenient when we want some automatic feature/variable selection or when dealing with highly correlated predictors

$$\min_{\beta} \frac{1}{n}(\mathbf{y} - X\beta)^T(\mathbf{y} - X\beta) + \lambda \sum_{j=1}^{d} |\beta_j|.$$

Singular Value Decomposition

- Unsupervised Machine Learning
  - no datasets are provided
- Can restructure data in a meaningful way
- A factorization of a matrix
  - $M = U\Sigma V$, where U and V are unitary matrices and Σ is a diagonal matrix of singular values.
- Best for:
  - Recommender systems
- Difficult to understand why data has been restructured
  - Not very transparent

Gathering Data

- We are grabbing data from the Tracktor website which has Amazon price histories.
- We are reading from a specific part of the html in order to get the data.
- We want your help with methods we can use to make sure our html page requests do not look like a DDOS attack.

Appeared DDOS Attacks?

- DDoS - distributed denial-of-service
  - attack occurs when multiple systems flood the bandwidth or resources of a targeted system, usually one or more web servers
  - Malicious
- Currently will not be a problem
  - We are not malicious
    - Educational purposes
  - May encounter this problem when scaling to more products