Machine Learning with Graphs (MLG)

RecSys: GBDT + LR

Pre-training cross features for recsys

Cheng-Te Li (李政德)

Institute of Data Science

National Cheng Kung University

chengte@mail.ncku.edu.tw

Sponsored Search: On Site Search

amazon prime

All michael kors

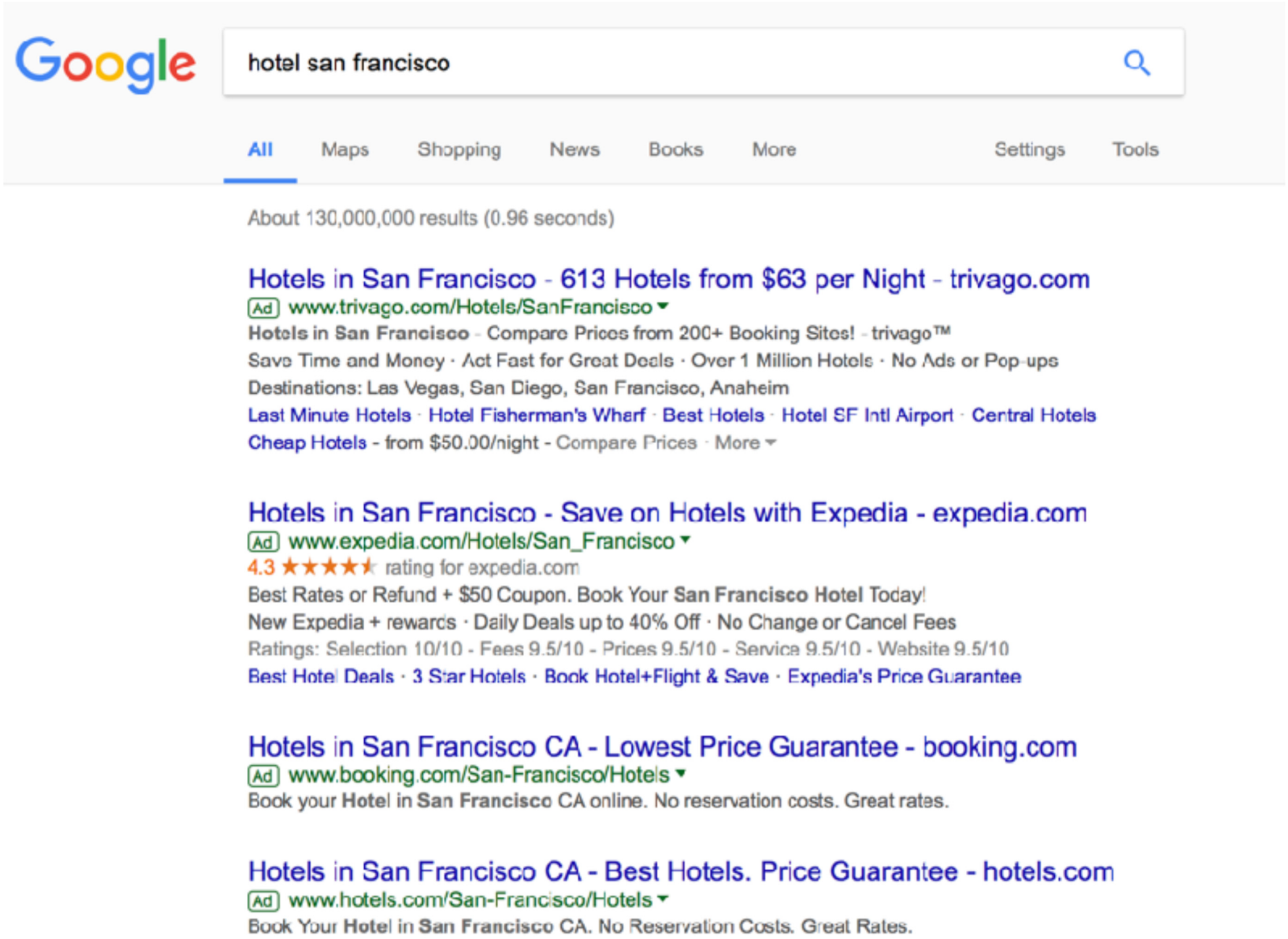
Departments Browsing History Mantrach's Amazon.com Today's Deals Gift Cards & Registry EN

Sponsored
Women RFID Blocking Wallet Genuine Leather Zip Around Clutch Large Travel Purse
$35.98 $99.98 prime
89

Sponsored
The Fix Mckenzie Suede and Leather Bucket Crossbody Bag
$119.00 prime
9

Sponsored
Yafeige Large Luxury Women's RFID Blocking Tri-fold Leather Wallet Zipper Ladies Clutch Purse
$28.86 $78.00 prime
438

Sponsored Search: Text Ads



Display Advertising

The New York Times

SUBSCRIBE NOW | SIGN IN | Register

INTERNATIONAL | DEALBOOK | MARKETS | ECONOMY | ENERGY | MEDIA | TECHNOLOGY | PERSONAL TECH | ENTREPRENEURSHIP

Exxon Mobil Investigated in New York Over Possible Lies on Climate

By JUSTIN GILLIS and CLIFFORD KRAUSS
3:30 PM ET

The sweeping inquiry, by the state attorney general, focuses on whether the oil company lied to the public and investors over the risks of climate change.

250 Comments

T. Fallon/Bloomberg, via Getty Images

An Exxon Mobil refinery in Los Angeles, Calif. The New York attorney general is investigating the oil and gas company.

LATEST NEWS

5:01 PM ET	'Grand Theft Auto' Maker Take-Two's Revenue Nearly Triples
5:00 PM ET	United Airlines CEO to Return in Early 2016 After Heart Attack
4:57 PM ET	NY Attorney General Investigating Exxon Over Climate Statements

MARKETS » At close 11/05/2015

European Union Predicts Economic Gains From Influx of Migrants

By JAMES KANTER
12:10 PM ET

Officials forecast that the three million arrivals expected by 2017 would provide a net gain of perhaps a quarter of 1 percent by that year to the European economy.

INSIGHT & ANALYSIS

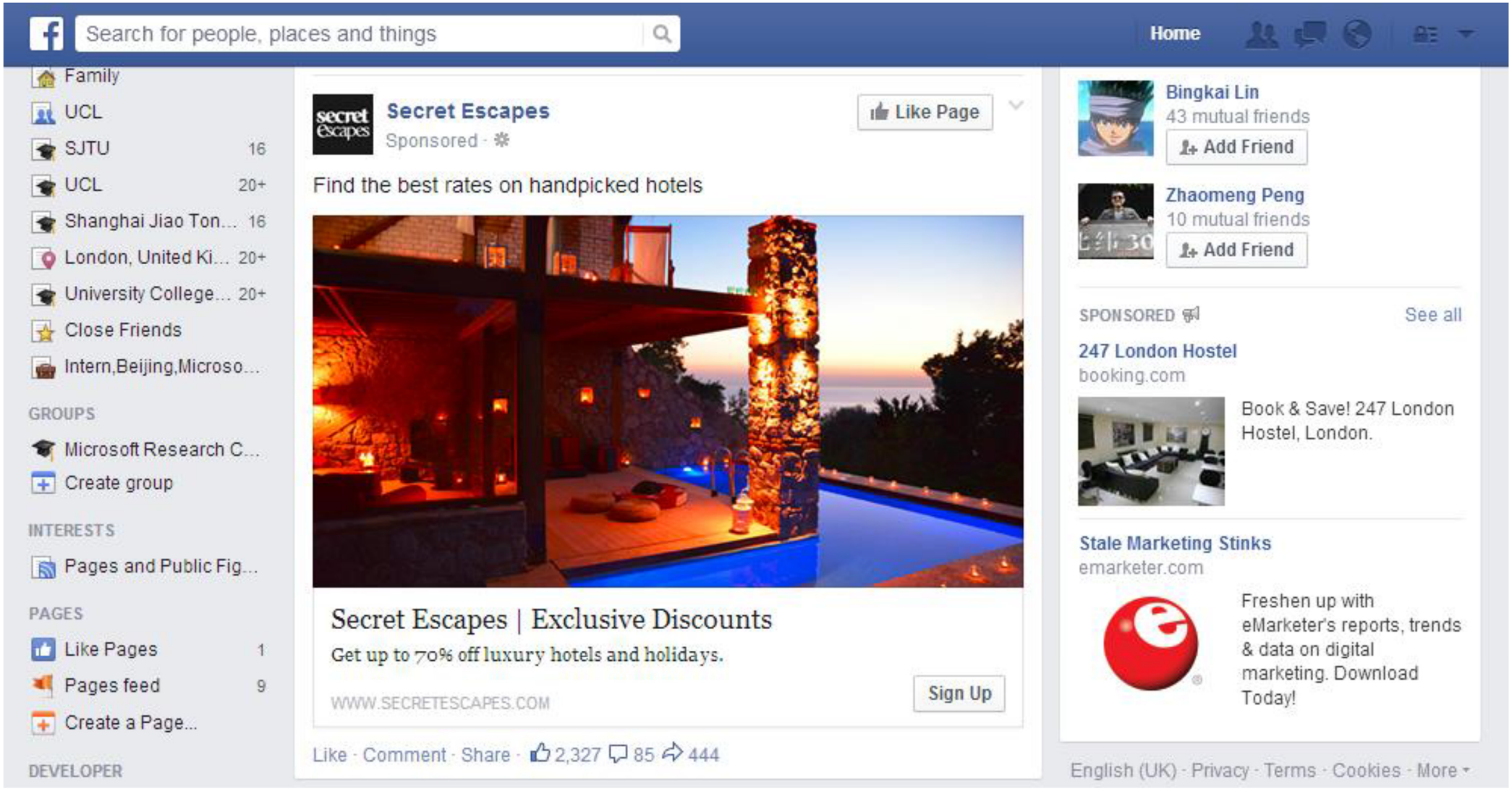
COMMON SENSE

Dewey Jury's Deadlock Exposes a System's Flaws

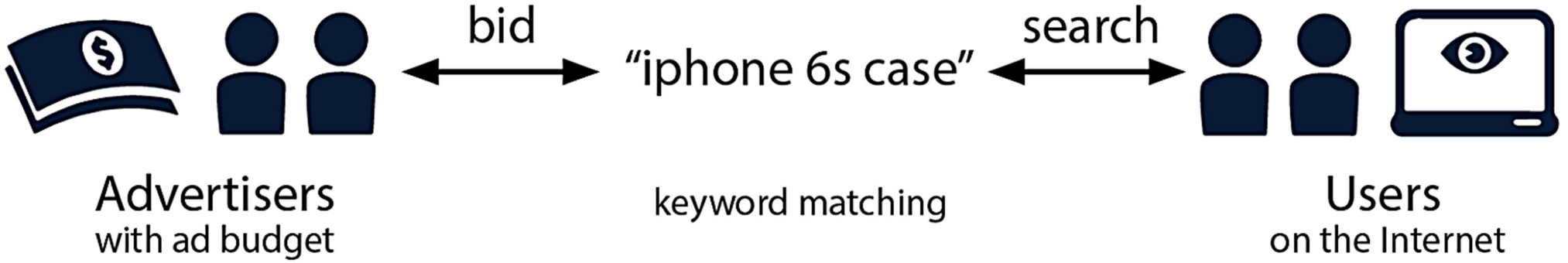
By JAMES B. STEWART
3:06 PM ET

One reason for the mistrial in the Dewey & LeBoeuf criminal case may have been the requirement for a unanimous decision.

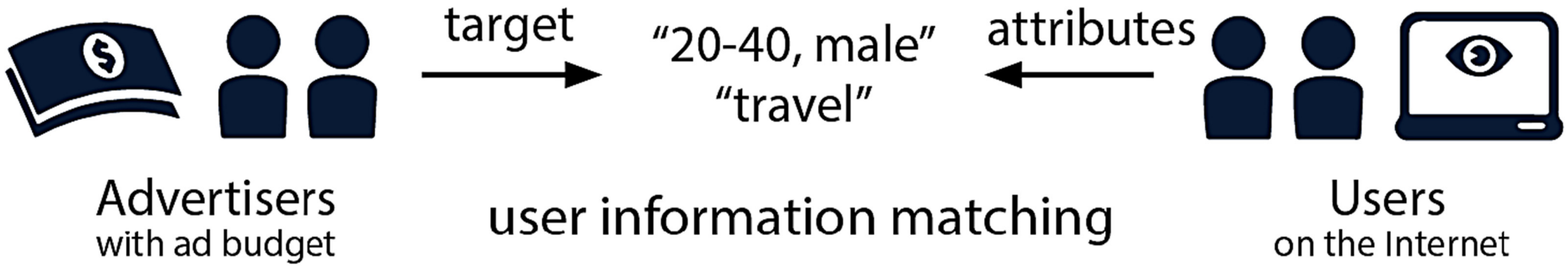
Display Advertising



Sponsored Search & Display Ads

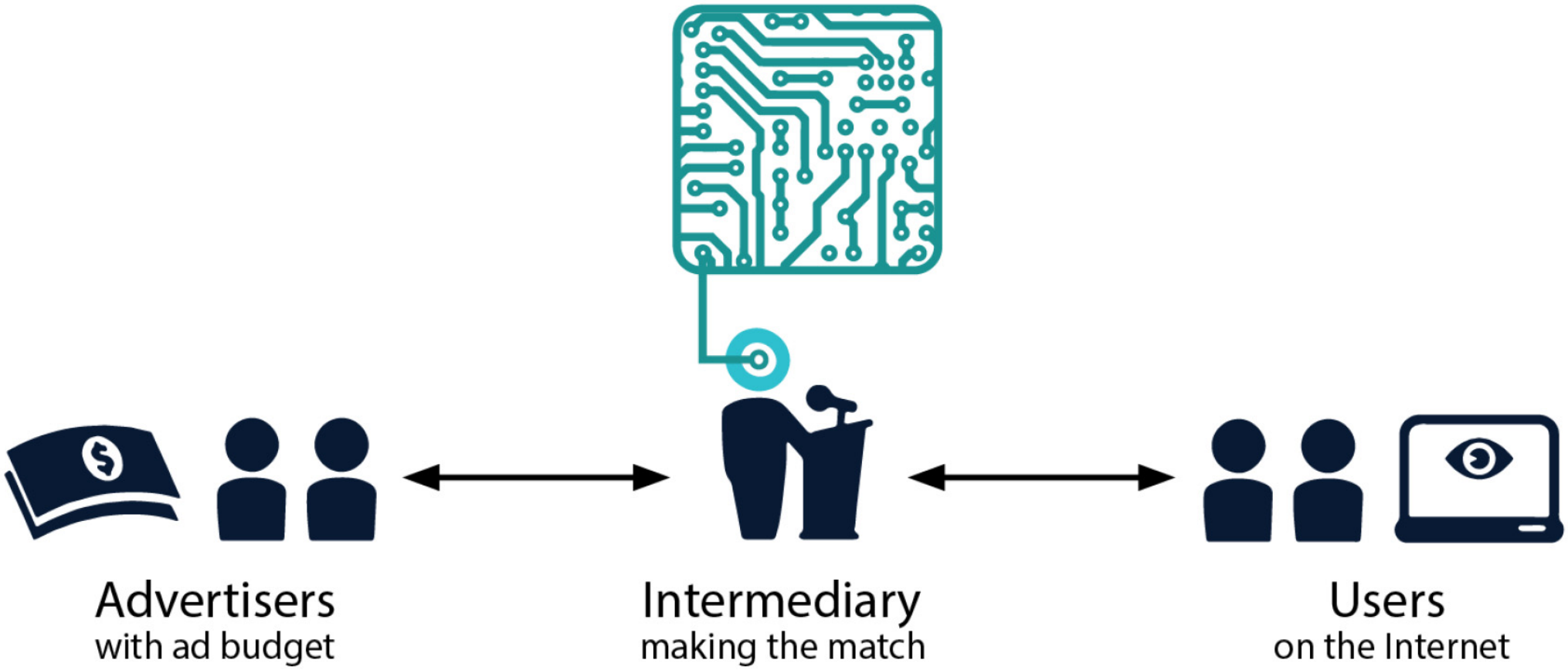


- Advertiser sets a bid price for the keyword
- User searches the keyword
- Search engine hosts the auction to ranking the ads



- Advertiser targets a segment of users
- Intermediary matches users and ads by user information

Computational Advertising



Design algorithms to make the best match between the advertisers and Internet users with economic constraints

Click-Through Rate (CTR)

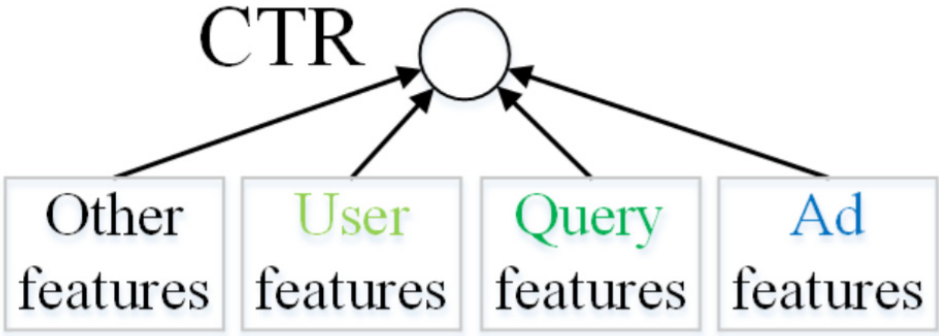
- CTR: ratio of users who click on an ad to the number of total users who view the ad

$$CTR = \frac{\#Clicks}{\#Impressions} \times 100\%$$

  - Measure the success of an online advertising campaign
  - Typical CTR is less than 1%
  - Revenue can be maximized by choosing to display ads that have the maximum CTR
- CTR prediction
  - **Predict conditional probability that the ad will be clicked by the user given features of user and ad**

CTR Prediction

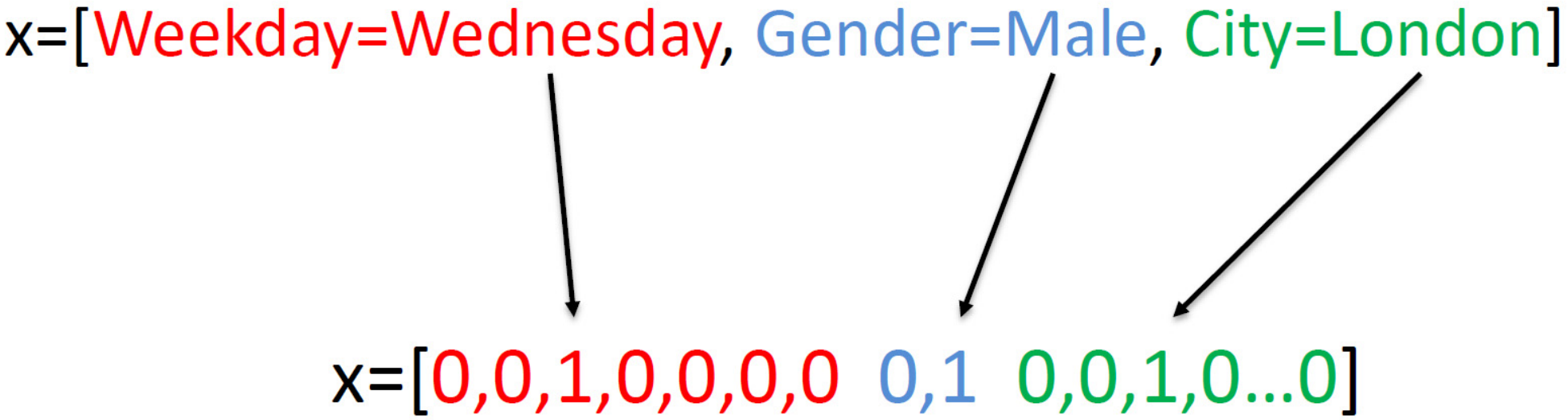
Date: 20160320
Hour: 14
Weekday: 7
IP: 119.163.222.*
Region: England
City: London
Country: UK
Ad Exchange: Google
Domain: yahoo.co.uk
URL: http://www.yahoo.co.uk/abc/xyz.html
OS: Windows
Browser: Chrome
Ad size: 300*250
Ad ID: a1890
User tags: Sports, Electronics
Ad tags: Tainan food, bubble milk tea
Ad's past day performance: 1.5%



Click (1) or not (0)?
Predicted CTR = 0.15

Feature Representation

- Binary one-hot encoding of categorical data



High dimensional sparse binary feature vector

Logistic Regression

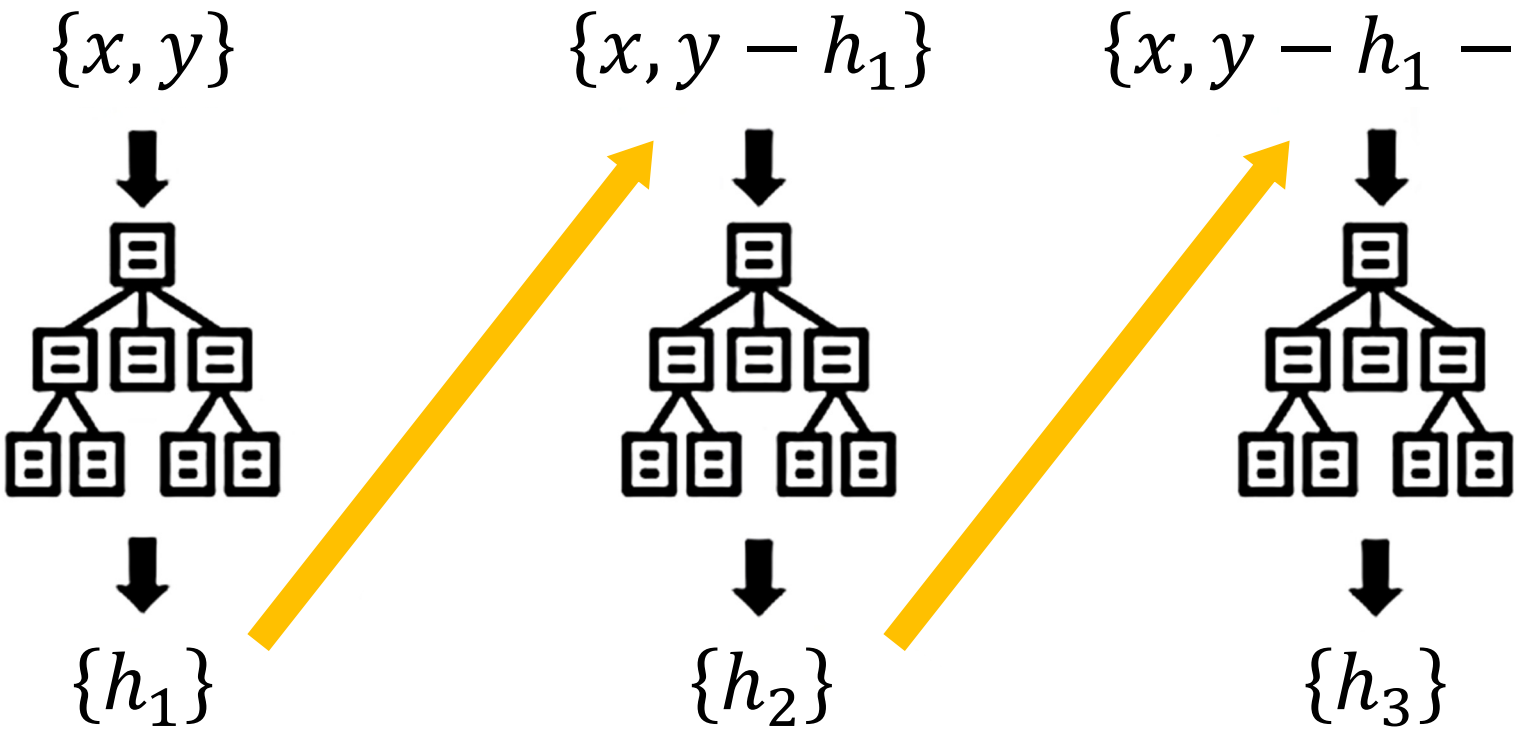
- A binary regression problem

$$\min_{\boldsymbol{w}} \sum_{(y,\boldsymbol{x})\in D} L(y,\hat{y}) + \lambda \cdot \|\boldsymbol{w}\|^2 \qquad \hat{y} = \frac{1}{1+\exp(-\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x})}$$

  - Cross-Entropy loss $L(y,\hat{y}) = -y\log\hat{y} - (1-y)\log(1-\hat{y})$
- However ...
  - Large binary feature space (e.g., > 10 millions)
  - Large data instance number (e.g., > 10 millions daily)
  - A seriously unbalanced label
    - Normally, #clicks/#non-clicks = 0.3%
    - Negative down sampling
  - Cannot capture the interactions between features
    - Need good feature engineering and feature selection

Gradient Boosted Decision Tree (GBDT)

Dataset: (x_i, y_i) for $i = 1$ to m



Every Prediction $h = h_1 + h_2 + h_3$

Sum of residuals

$$E = \sum (y - h)^2$$

Gradient descent

$$\frac{\partial E}{\partial h} = 2\sum (y - h)(-1) \qquad h = h - \alpha \cdot \left(\frac{\partial E}{\partial h}\right) = h + \alpha \cdot \sum (y - h)$$

Gradient Boosted Decision Tree (GBDT)

- Gradient boosting ML
  - For regression and classification
  - Produce a prediction model
  - Ensemble of weak prediction models, typically decision trees
- MSE as loss function
  - Minimize L by gradient descent to update the predictions y_i^p

$$L = MSE = \sum (y_i - y_i^p)^2$$

y_i: the ground-truth value of instance i
y_i^p: the predicted value of instance i

$$y_i^p = y_i^p - \alpha \cdot \left(\frac{\partial L}{\partial y_i^p}\right) = y_i^p + \alpha \cdot 2 \cdot \sum (y_i - y_i^p)$$

$\sum (y_i - y_i^p)$: sum of prediction residuals, α: learning rate

Update the predictions such that the sum of our residuals is close to 0 (or minimum) and predicted values are sufficiently close to actual values

Boosting is Playing Golf

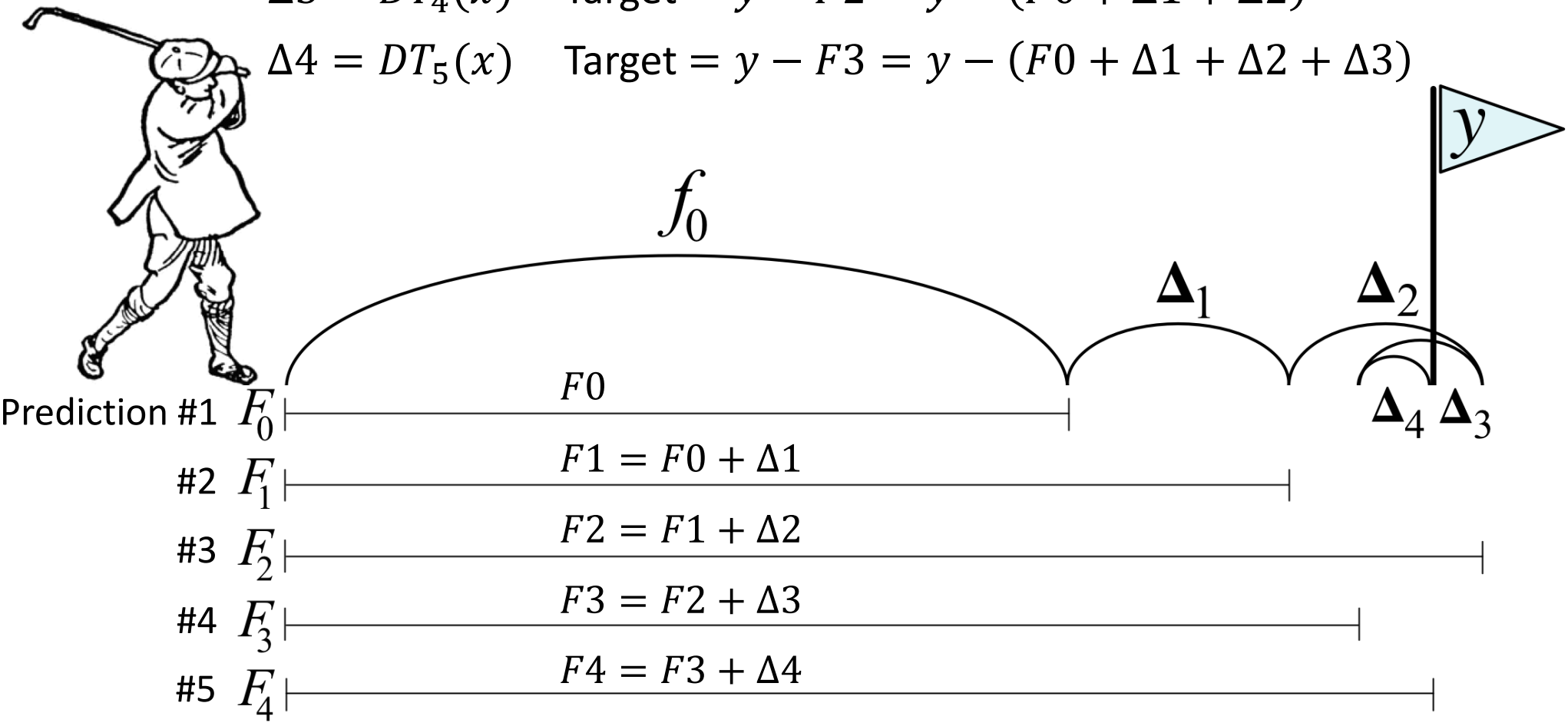
$F0 = DT_1(x)$ Target $= y$

$\Delta 1 = DT_2(x)$ Target $= y - F0$

$\Delta 2 = DT_3(x)$ Target $= y - F1 = y - (F0 + \Delta 1)$

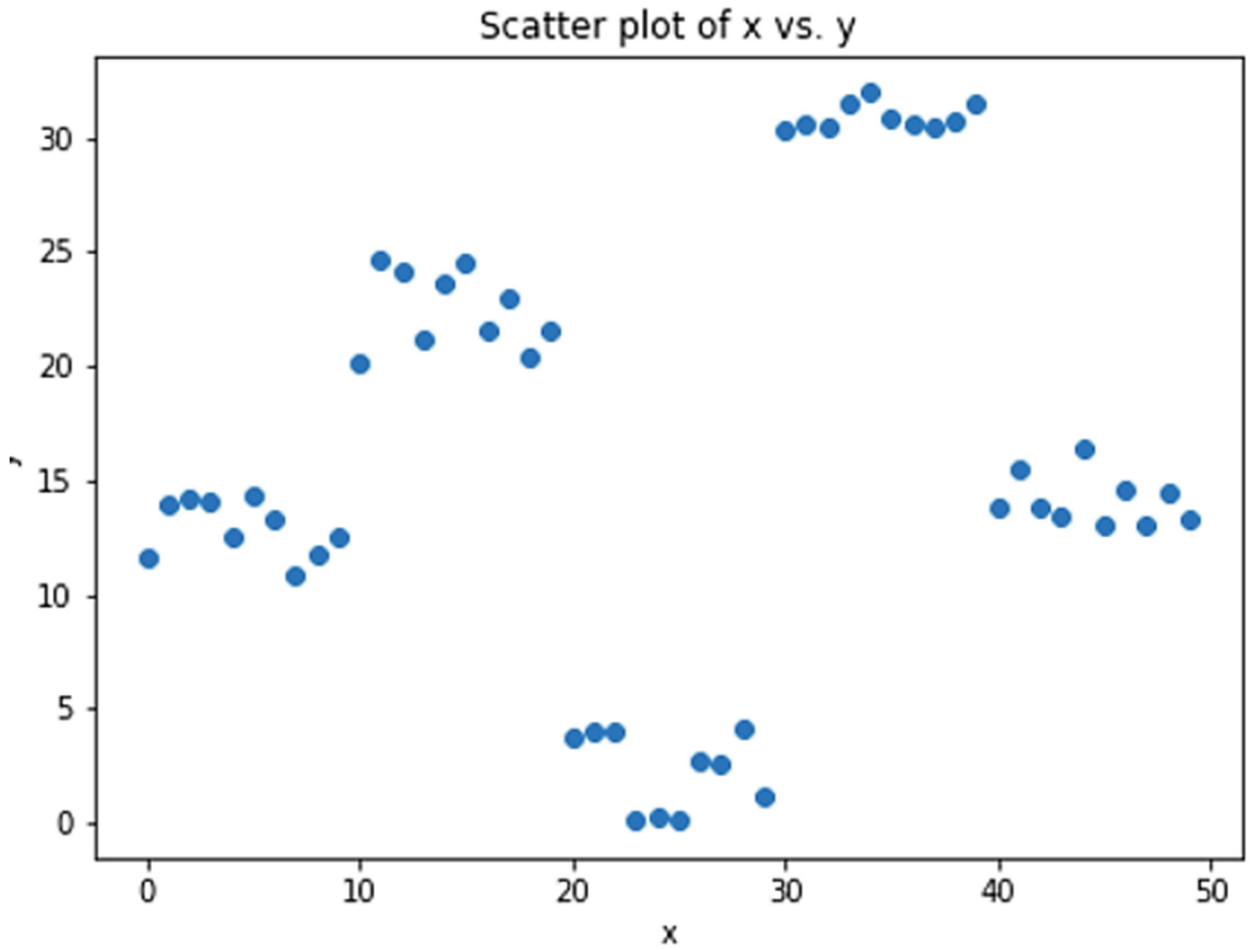
$\Delta 3 = DT_4(x)$ Target $= y - F2 = y - (F0 + \Delta 1 + \Delta 2)$

$\Delta 4 = DT_5(x)$ Target $= y - F3 = y - (F0 + \Delta 1 + \Delta 2 + \Delta 3)$



GBDT Steps

Let's consider simulated data as shown in scatter plot below with 1 input (x) and 1 output (y) variables



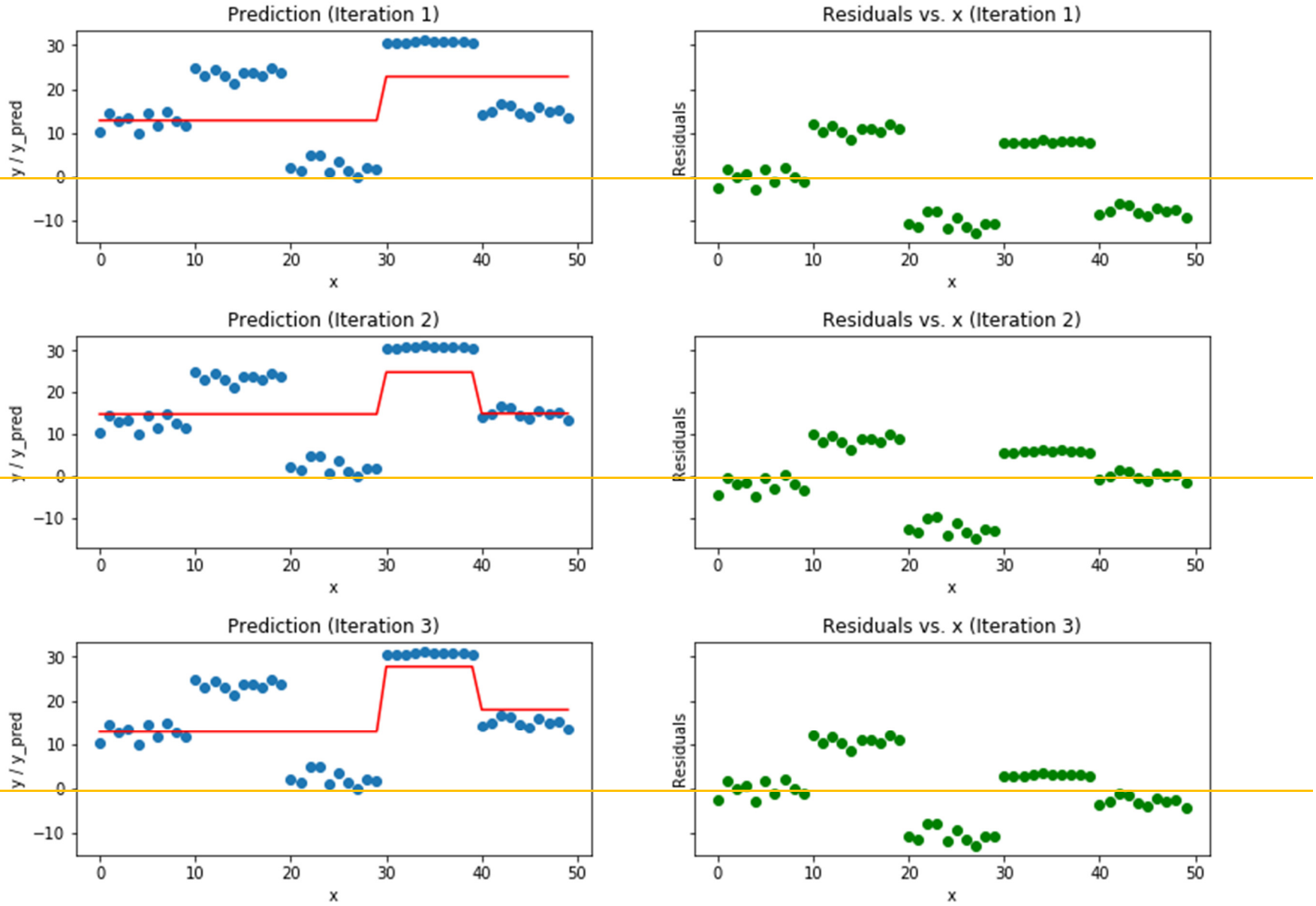
GBDT Steps

- **Step 1:** Fit a simple decision tree on data (Input = x, output = y)
- **Step 2:** Calculate error residuals $e_1 = y - \hat{y}_1$
  - i.e., actual target value, minus predicted target value
  - $\hat{y}_1 = DT_1(x)$
- **Step 3:** Fit a new model on error residuals as prediction target with the same feature (i.e., x)
  - i.e., fit model that considers e_1 as prediction targets
  - $\hat{e}_1 = DT_2(x)$
- **Step 4:** Add predicted residuals to previous predictions
  - $\hat{y}_2 = \hat{y}_1 + \hat{e}_1$
- **Step 5:** Fit another model on residuals that is still left and repeat steps 2-5 until it starts overfitting or the sum of residuals become constant
  - $e_2 = y - \hat{y}_2, \hat{e}_2 = DT_3(x), \hat{y}_3 = \hat{y}_2 + \hat{e}_2$
  - $e_3 = y - \hat{y}_3, \hat{e}_3 = DT_4(x), \hat{y}_4 = \hat{y}_3 + \hat{e}_3$, and so on

Overfitting can be controlled by consistently checking accuracy on validation data

Exampled GBDT Visualization

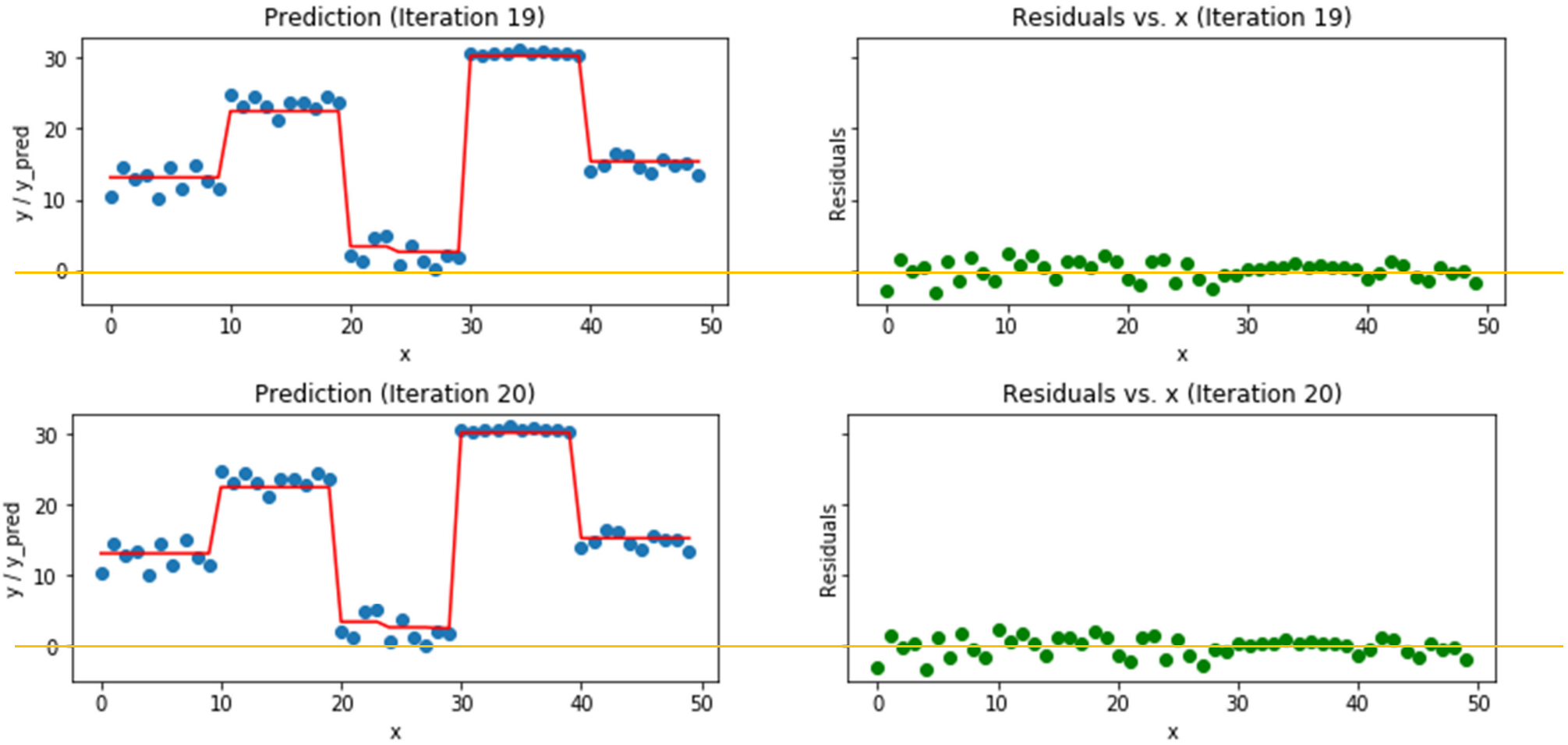
https://www.kaggle.com/grroverpr/gradient-boosting-simplified/



Exampled GBDT Visualization

- After 20th iteration, residuals are randomly distributed around 0
- The predictions are very close to true values
- Note: #iterations are called **n_estimators** in sklearn
- This would be a good point to stop or our model will start overfitting

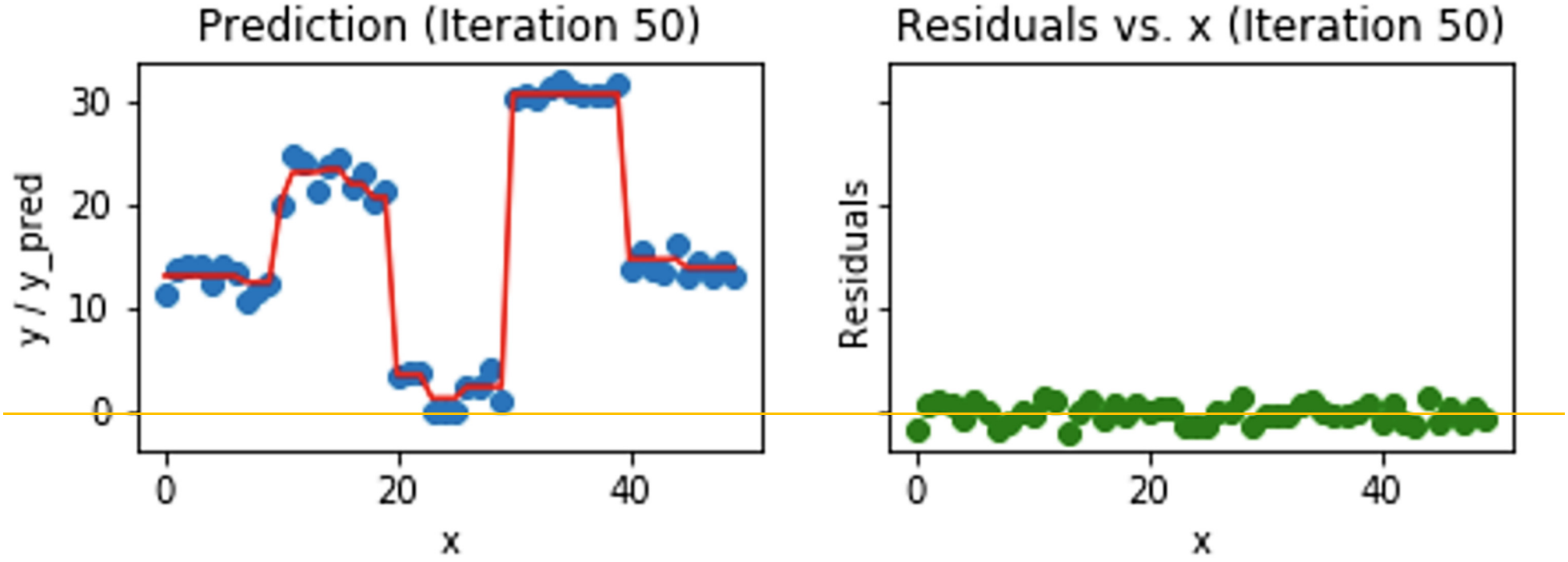
https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingClassifier.html



Exampled GBDT Visualization

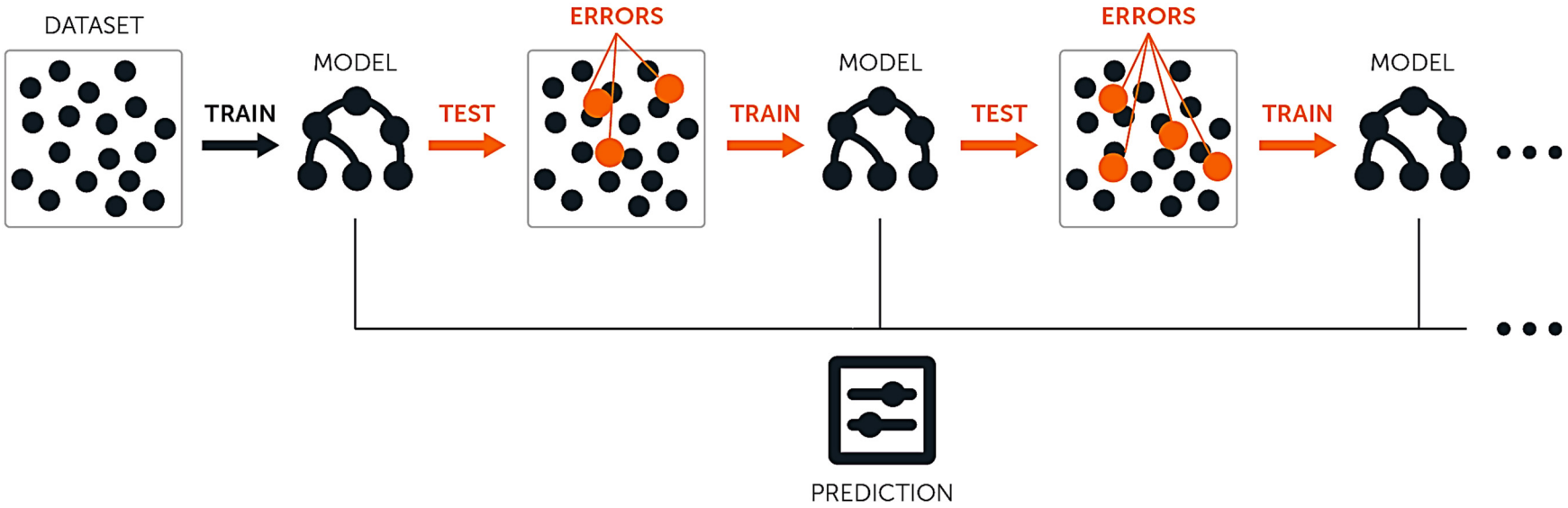
What will the model look like for 50th iteration?

- Residuals vs. x plot look similar to what we see at 20th iteration
- But the model is becoming more complex
- Predictions are overfitting on training data (trying to fit each training data)
- So, it would have been better to stop at 20th iteration

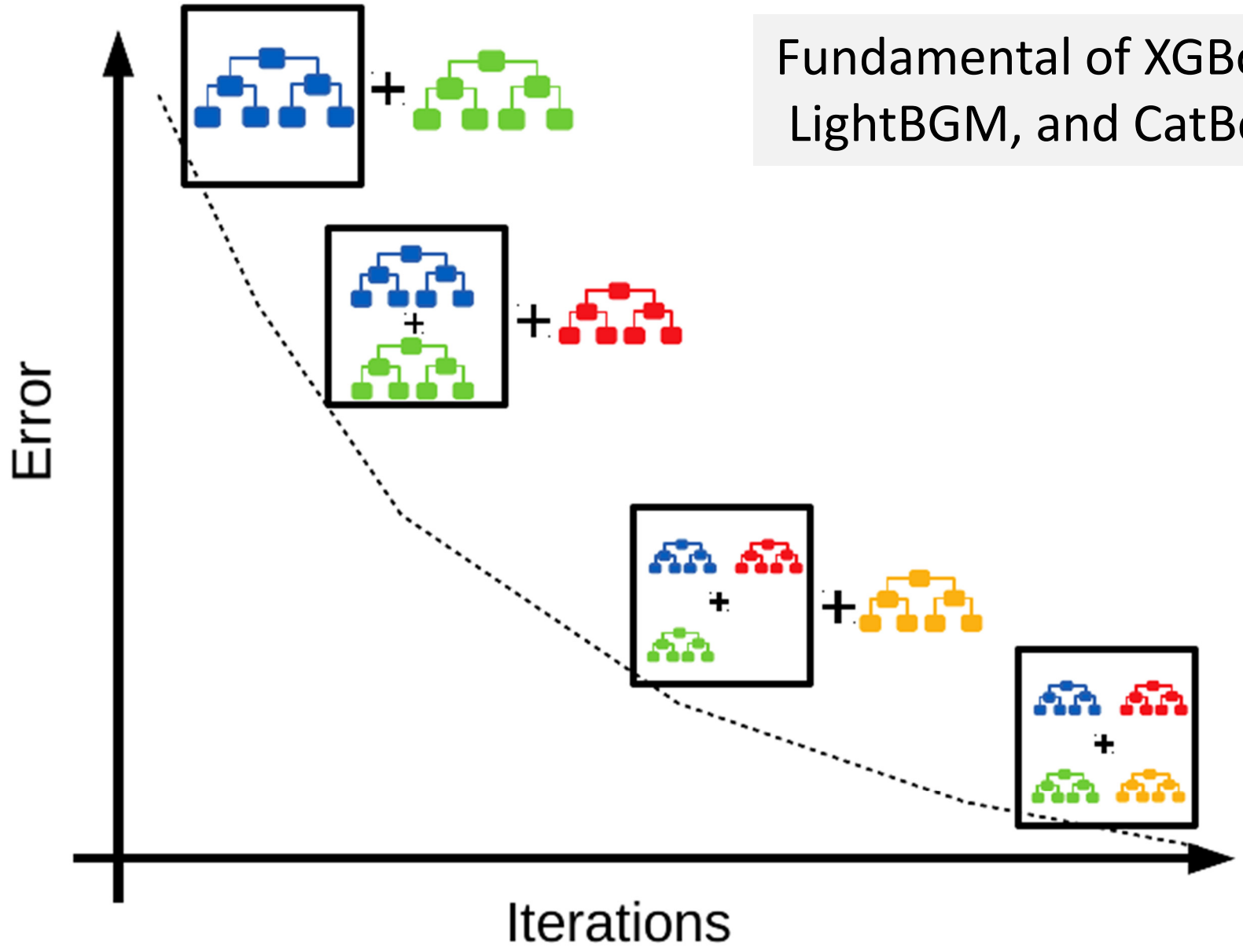


Intuition behind GBDT

- Repetitively leverage the patterns in residuals and strengthen a model with weak predictions and make it better
  1) First model data with simple models and analyze errors
  2) **Errors signify difficult-to-fit data points** by simple model
     i.e., identify wrong predictions
  3) Later models particularly focus on those hard to fit data
     → get these difficult data points right
  4) Combine all predictors by giving weights to each predictor

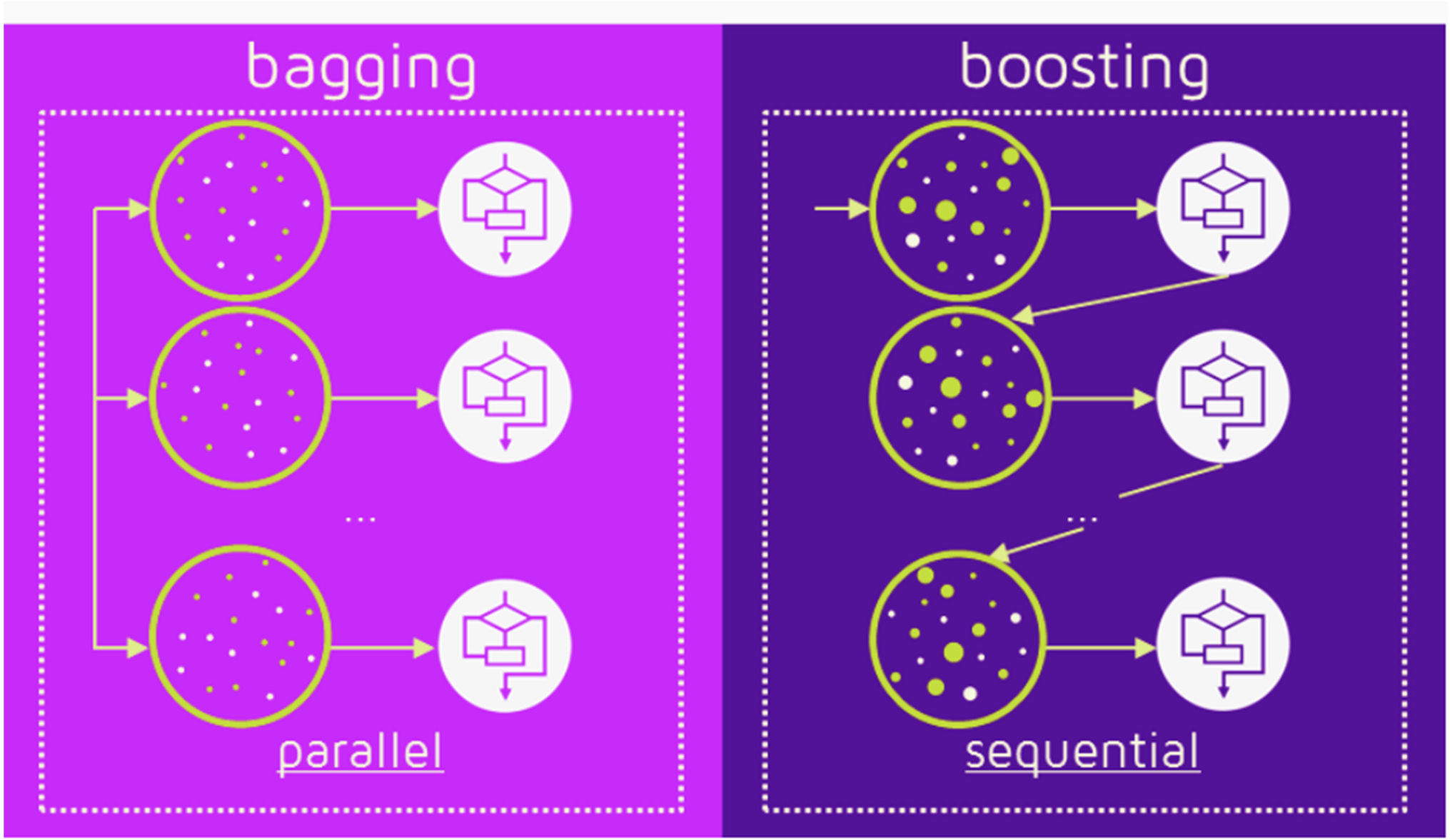


Summary of GBDT: Additive DTs for Prediction



Fundamental of XGBoost, LightBGM, and CatBoost

Bagging vs. Boosting



CTR Prediction / Ad Recommendation

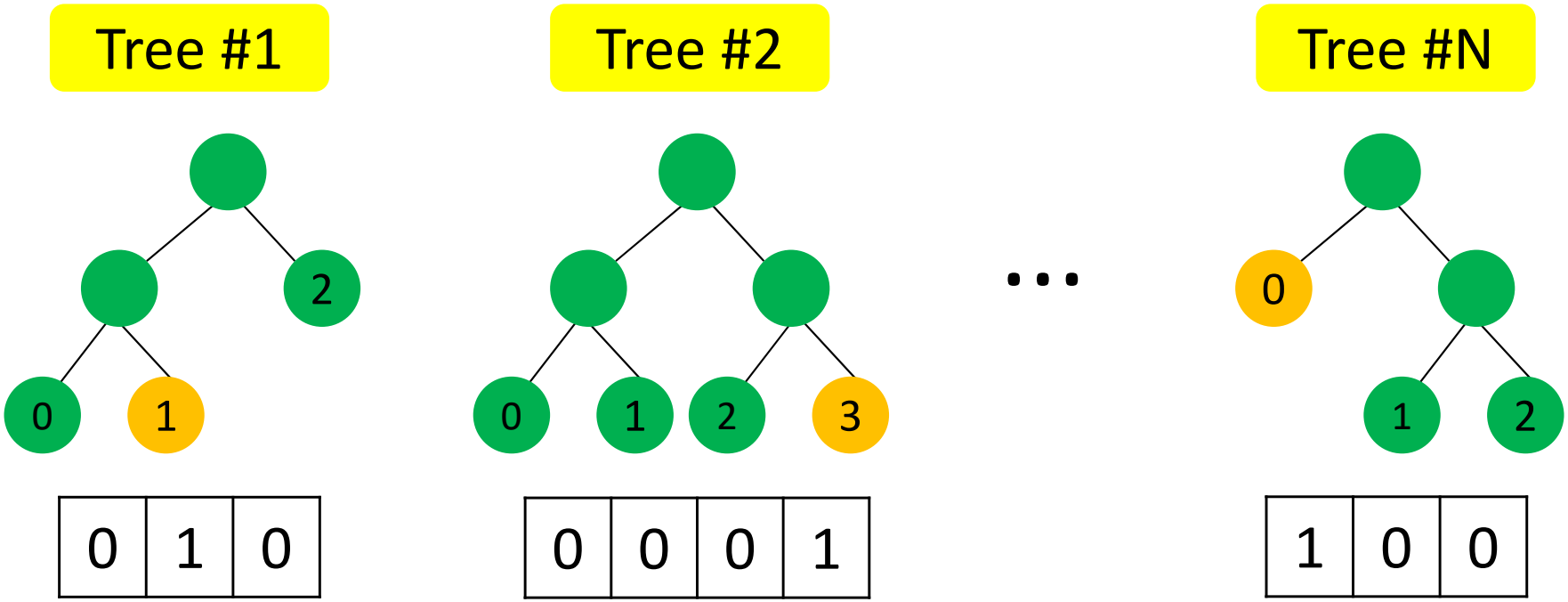
- Recall: $\mathbf{x} = [\mathbf{x}_u, \mathbf{x}_i, \mathbf{f}_u, \mathbf{f}_i], y \in \{0, 1\}$
  - $\mathbf{x}_u, \mathbf{x}_i$: one-hot encodings
  - $\mathbf{f}_u, \mathbf{f}_i$: multi-hot encodings
  - Numerical feat → Binning (non-linear feature transformation)
  - Categorical feat → Cartesian product (tuple input transformation)

$$A = \{a, b\}, B = \{0,1,2\}$$
$$A \times B = \begin{Bmatrix}(a,0),(a,1),(a,2),\\(b,0),(b,1),(b,2)\end{Bmatrix}$$

- Feature interaction is crucial
  - Users' past items + item attributes + user attributes
- Cross features
  - ["20 < age < 25" & "1K < price < 2K" & "color = blue"]
  - ["gender = female" & "device = iphone 8" & "city = Tainan" & "time = 8pm" & "price > 5K" & "last item = cake"]
- How to automatically obtain effective cross features?

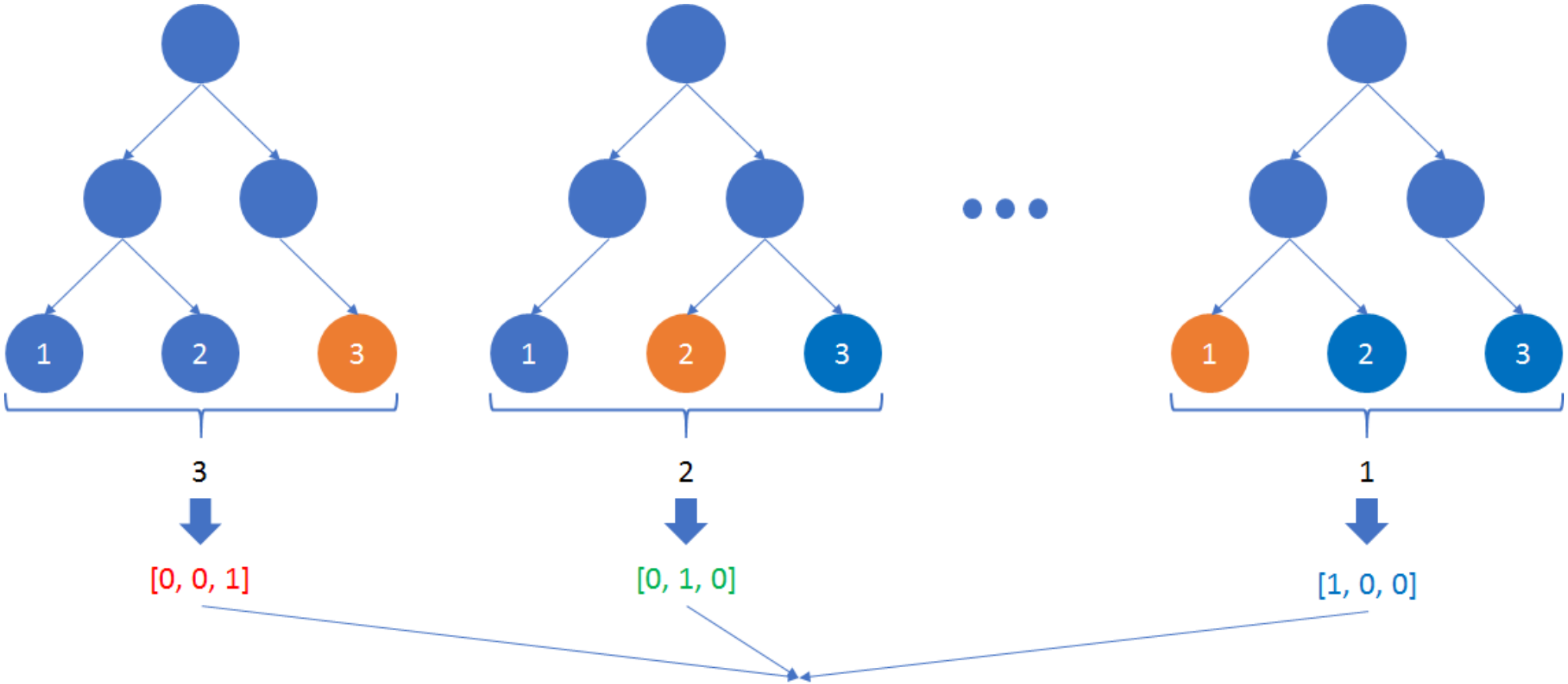
Feature Transformation by GBDT

- Use the pre-trained GBDT to obtain cross features
  - Each leaf in a tree generates a cross feature
  - Use one-hot encoding to represent each cross feature
- Generate cross features to be a multi-hot encoding



["20 < age < 25" & "1K < price < 2K" & "color = blue"]

GBDT + LR



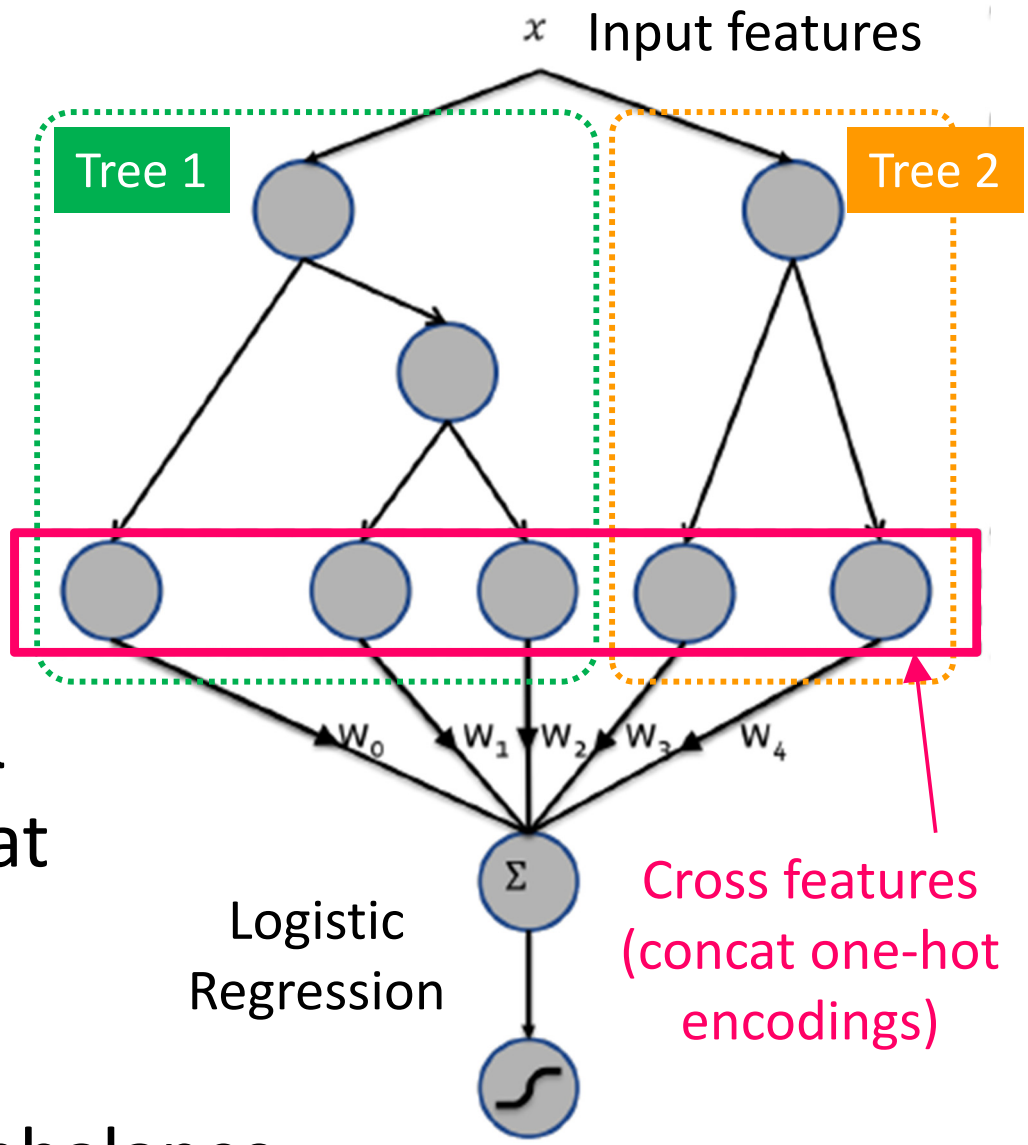
$$\boldsymbol{x} = [0, 0, 1, 0, 1, 0, \ldots, 1, 0, 0]$$

$$\min_{\boldsymbol{w}} \sum_{(y,\boldsymbol{x})\in D} L(y, \hat{y}) + \lambda \cdot \|\boldsymbol{w}\|^2 \qquad \hat{y} = \frac{1}{1 + \exp(-\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x})}$$

$$L(y, \hat{y}) = -y \log \hat{y} - (1 - y) \log(1 - \hat{y})$$

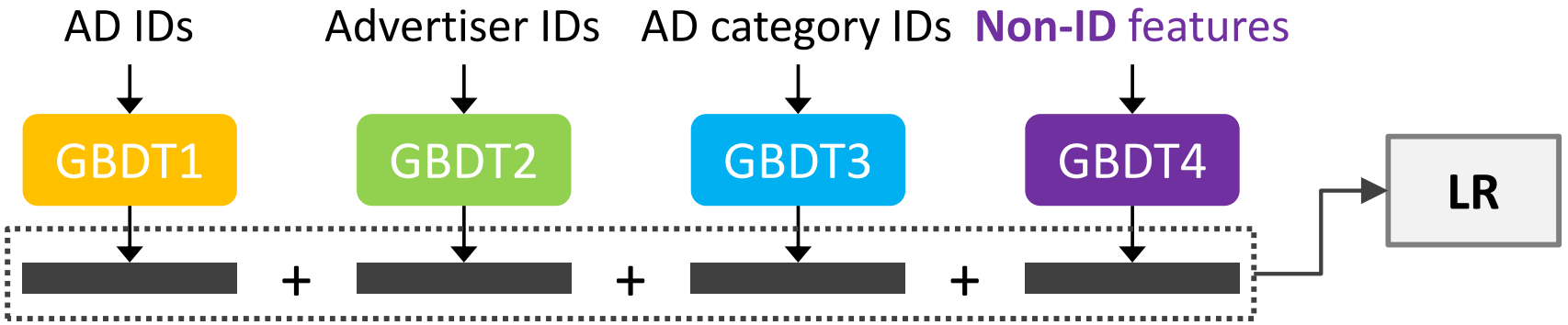
GBDT + LR: Insights

- Supervised feature encoding
  - Real values → Binary vectors
- Non-linear transformation
  - From root to leaf
- Rules bring explainability
  - Reflected by LR weights
- #CrossFeat = #Trees
- Feature selection by GBDT & removing irrelevant CrossFeat
- Do NOT forget:
  negative down sampling
  to deal with extreme data imbalance



Reflection on GBDT+LR

- **Q1:** Why adopting GBDT, not random forest?
  - Boosting in GBDT lets some trees focus on difficult instances
  - RF does not deal with residuals
  - Every training instance enters into all trees in GBDT
  - Not every instance enters into all trees in RF
- **Q2:** Should we consider item ID and user ID in GBDT?
  - Yes, absolutely! But item popularity is usually power-law
  - → infeasible to combine multi-hot ID vector with other features
  - Solution: power-law and non-power-law features by multi GBDTs



Reflection on GBDT+LR

- **Q3:** Can original feats be combined with cross feats?
  - Yes. You can try it
- **Q4:** How to set #Trees and tree depth?
  - Trade-off between accuracy and efficiency
  - Hyperparameter tuning (e.g., #Trees = 500)
- **Q5:** Why logistic regression?
  - The scale of user-item interactions is massive
  - LR is good at efficiency
  - You can try GBDT with NN and FM, along with BPR loss
  - GBDT can be replaced with XGBoost, LightGBM, and CatBoost
- **Q6:** What is the strength of GBDT+LR?
  - Learn effective features ↔ embedding learning in DL-based RecSys

References / Code

- J. H. Friedman. "**Greedy function approximation: A gradient boosting machine**" Annals of Statistics, 29(5), 2001 9802 cites
  - https://blog.csdn.net/w28971023/article/details/8240756
  - https://www.kaggle.com/grroverpr/gradient-boosting-simplified/
  - https://sefiks.com/2018/10/04/a-step-by-step-gradient-boosting-decision-tree-example/
  - https://sefiks.com/2018/10/29/a-step-by-step-gradient-boosting-example-for-classification/
  - https://explained.ai/gradient-boosting/index.html
  - http://tvas.me/articles/2019/08/26/Block-Distributed-Gradient-Boosted-Trees.html

- X. He et al. "**Practical Lessons from Predicting Clicks on Ads at Facebook**" ADKDD 2014 372 cites
  - GBDT+LR Code: https://github.com/wzhe06/CTRmodel