BT5152 Tutorial 5

AY 2018/19, Semester 1, Week 7

Lu Wei

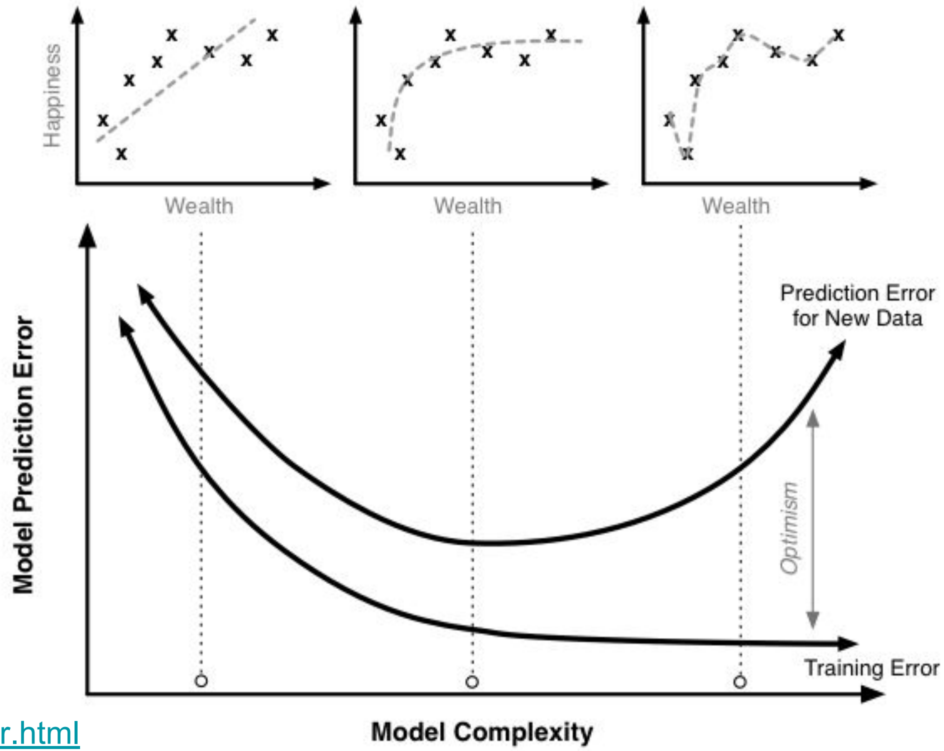
Assignment 1 Review

- Make sure your code is runnable.
  - Avoid typos, commented out code, wrong code.
  - setwd(), read.csv('/Users/luwei/workspace/data.csv'): no no no
- Discussions, how not to lose marks?
  - Do not state the obvious. Discuss the why, not the what. e.g. why test accuracy is low while train accuracy is high? What could be the possible cause?
- The order of min-max scaling & one-hot, when both need to be applied
- Clean code:
  - DRY: https://en.wikipedia.org/wiki/Don%27t_repeat_yourself (e.g. extract into functions)
  - Meaningful variable names
  - Avoid magic numbers. e.g. prefer column names over column indexes
  - When addressing a type of columns (e.g. numeric/categorical), do not list columns
  - Comments should explain why, not what
  - Less is more: do your data exploration but don't include it in submitted code

Key Concepts Revision

Recall Bias Variance Trade-off & Boosting

- Overfitting vs. Underfitting
- Boosting:
    - Weighted bootstrapping + averaging (for regression) / voting (for classification)
    - Likely to overfit
    - Slow due to sequential operations



Graph from: http://scott.fortmann-roe.com/docs/MeasuringError.html

Regularization in Linear Regression

$$J(\theta) = \frac{1}{2m}\left[\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2 + \lambda \sum_{j=1}^{n} \theta_j^2\right]$$

$$\min_\theta J(\theta)$$

- $\lambda \sum_{j=1}^{n} \theta_j^2$ Intuition: to reduce the effect of ALL features in the model
- Helps reduce bias or variance?

Source: https://www.coursera.org/lecture/machine-learning/regularized-linear-regression-QrMXd

Regularization in XGBoost

$$\mathcal{L}(\phi) = \sum_i l(\hat{y}_i, y_i) + \sum_k \Omega(f_k)$$

$$\text{where } \Omega(f) = \gamma T + \frac{1}{2}\lambda \|w\|^2$$

- What’s w in the context of a decision tree?
- What’s T?

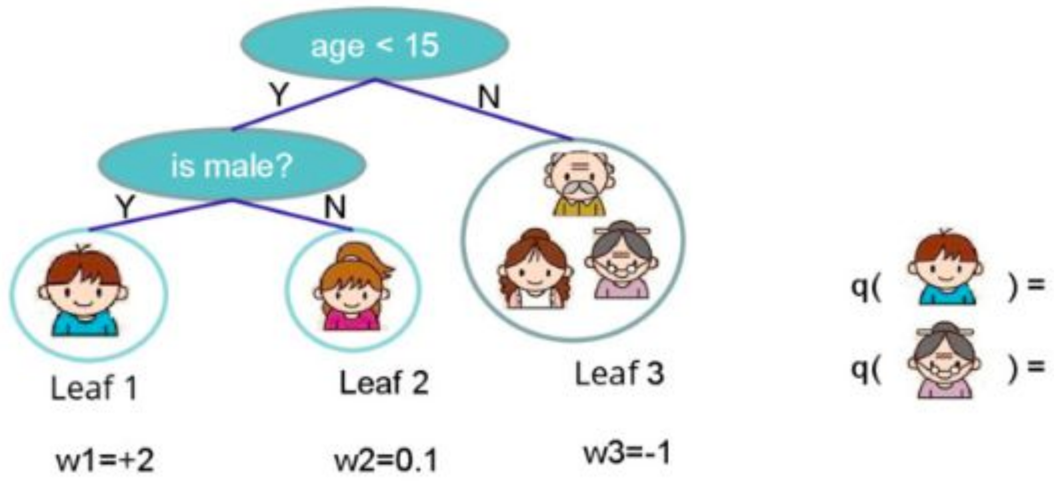
L1 vs. L2 Regularization

- L1 aka Lasso Regression
- L2 aka Ridge Regression
- $|w|$ vs. w^2
- Why is L1 good for sparse/high dimensional features?

Reference: https://towardsdatascience.com/l1-and-l2-regularization-methods-ce25e7fc831c

Examples

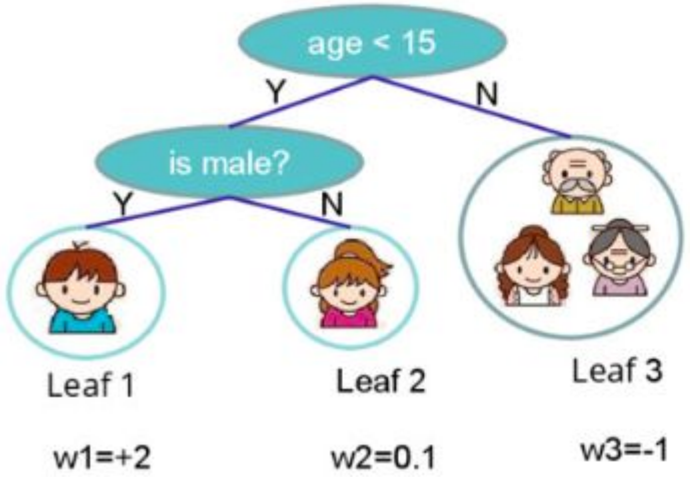
$$\mathcal{F} = \{f(\mathbf{x}) = w_{q(\mathbf{x})}\}(q : \mathbb{R}^m \rightarrow T, w \in \mathbb{R}^T)$$



Source: http://datascience.la/xgboost-workshop-and-meetup-talk-with-tianqi-chen/

Examples

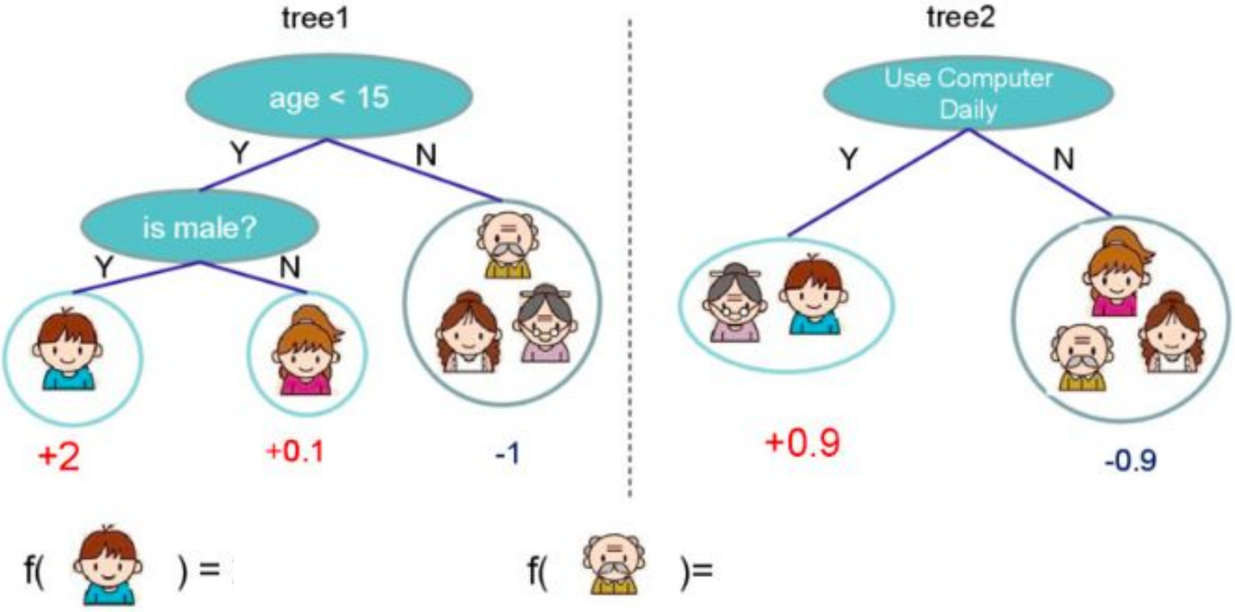
$$\Omega(f) = \gamma T + \frac{1}{2}\lambda \|w\|^2$$



$$\Omega =$$

Examples

$$\hat{y}_i = \phi(\mathbf{x}_i) = \sum_{k=1}^{K} f_k(\mathbf{x}_i), \quad f_k \in \mathcal{F},$$



Split Finding

$$\mathcal{L}(\phi) = \sum_i l(\hat{y}_i, y_i) + \sum_k \Omega(f_k)$$

$$\text{where } \Omega(f) = \gamma T + \frac{1}{2}\lambda \|w\|^2$$

$$\xrightarrow[\text{Calculus}]{\text{Taylor expansion}}$$

$$w_j^* = -\frac{G_j}{H_j + \lambda}$$

$$\text{Loss*} = -\frac{1}{2}\sum_{j=1}^{T} \frac{G_j^2}{H_j + \lambda} + \gamma T$$

$$Gain = \frac{1}{2}\left[\frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{(G_L + G_R)^2}{H_L + H_R + \lambda}\right] - \gamma$$

For more maths, see: https://xgboost.readthedocs.io/en/latest/tutorials/model.html

XGBoost

- Why so fast?
  - Parallel tree boosting **within** each tree at independent branch level
  - Written in C++ with smart memory management
- Handles missing values – no imputation needed
- Provides feature importance analysis
- Can still overfit
- Doesn’t perform feature engineering for you

Tutorial Exercises:

RStudio > Console:

```
# install.packages("swirl")
library(swirl)
# delete_progress('your name')
install_course_github('weilu', 'BT5152', multi=TRUE)
swirl()
```

1: XGBoost