Data Mining Techniques

CS 6220 - Section 3 - Fall 2016

Lecture 21: Review

Jan-Willem van de Meent

Schedule

13	30 Nov	Bonus Topic: Deep Learning	#4 due	
	02 Dec	Review		
14	07 Dec	(No Class)		
	09 Dec	Final Exam		
15	14 Dec	Project Presentations		Reports due
16	19 Dec	(Final grades posted)		

Topics for Exam

Pre-Midterm

- Probability
- Information Theory
- Linear Regression
- Classification
- Clustering

Post-Midterm

- Topic Models
- Dimensionality Reduction
- Recommender Systems
- Association Rules
- Link Analysis
- Time Series
- Social Networks

Post-Midterm Topics

Topic Models

- Bag of words representations of documents
- Multinomial mixture models
- Latent Dirichlet Allocation
  - Generative model
  - Expectation Maximization (PLSA/PLSI)
  - Variational inference (high level)
- Perplexity
- Extensions (high level)
  - Dynamic Topic Models
  - Supervised LDA
  - Ideal Point Topic Models

Dimensionality Reduction

Principal Component Analysis

- Interpretation as minimization of reconstruction error
- Interpretation as maximization of captured variance
- Interpretation as EM in generative model
- Computation using eigenvalue decomposition
- Computation using SVD
- Applications (high-level)
  - Eigenfaces
  - Latent Semantic Analysis
    - Relationship to LDA
  - Multi-task learning
- Kernel PCA
  - Direct method vs modular method

Dimensionality Reduction

- **Canonical Correlation Analysis**
  - Objective
  - Relationship to PCA
  - Regularized CCA
    - Motivation
    - Objective
- **Singular Value Decomposition**
  - Definition
  - Complexity
  - Relationship to PCA
- **Random Projections**
  - Johnson-Lindenstrauss Lemma

Dimensionality Reduction

- **Stochastic Neighbor Embeddings**
  - Similarity definition in original space
  - Similarity definition in lower dimensional space
  - Definition of objective in terms of KL divergence
  - Gradient of objective

Recommender Systems

- Motivation: The long tail of product popularity
- Content-based filtering
  - Formulation as a regression problem
  - User and item bias
  - Temporal effects
- Matrix Factorization
  - Formulation of recommender systems as matrix factorization
  - Solution through alternating least squares
  - Solution through stochastic gradient descent

Recommender Systems

- **Collaborative filtering**
  - (user, user) vs (item, item) similarity
    - pro’s and cons of each approach
  - Parzen-window CF
  - Similarity measures
    - Pearson correlation coefficient
      - Regularization for small support
      - Regularization for small neigborhood
    - Jaccard similarity
      - Regularization
    - Observed/expected ratio
      - Regularization

Association Rules

- Problem formulation and examples
  - Customer purchasing
  - Plagiarism detection
- Frequent Itemset
  - Definition of (fractional) support
- Association Rules
  - Confidence
  - Measures of interest
    - Added value
    - Mutual information

Association Rules

- **A-priori**
  - Base principle
  - Algorithm
  - Self-joining and pruning of candidate sets
  - Maximal vs closed itemsets
  - Hash tree implementation for subset matching
  - I/O and memory limited steps
  - PCY method for reducing candidate sets
- **FP-Growth**
  - FP-tree construction
  - Pattern mining using conditional FP-trees
- Performance of A-priori vs FP-growth

Aside: PCY vs PFP (parallel FP-Growth)

I asked an actual expert

Matteo Riondato

I notice that Spark MLib ships PFP as its main algorithm and I notice you benchmark against this as well. That said I can imagine there are might be different regimes where these algorithms are applicable. For example I notice you look at large numbers of transactions (order 10^7) but relatively small numbers of frequent items (10^3-10^4). The MMDS guys seem to emphasize the case where you cannot hold counts for all candidate pairs in memory, which presumably means numbers of items of order (10^5-10^6). Is it the case that once you are doing this at Walmart or Amazon scale, you in practice have to switch to PCY-variants?

Hi Jan,

This is a good question.

In my opinion, it is not true that if you have million of items then you need to use PCY-variants. FP-Growth and its many of variants are most likely going to perform better anyway, because available implementations have been seriously optimized. They are not really creating and storing pairs of candidates anyway, so that's not really the problem.

Hope this helps,

Matteo

PARMA: a parallel randomized algorithm for approximate association rules mining in MapReduce
M Riondato, JA DeBrabant, R Fonseca… - Proceedings of the 21st …, 2012 - dl.acm.org
Abstract Frequent Itemsets and Association Rules Mining (FIM) is a key task in knowledge discovery from data. As the dataset grows, the cost of solving this task is dominated by the component that depends on the number of transactions in the dataset. We address this issue
Cited by 68   Related articles   All 16 versions   Cite   Save

Link Analysis

- Recursive formulation
  - Interpretation of links as weighted votes
  - Interpretation as equilibrium condition in population model for surfers (inflow equal to outflow)
  - Interpretation as visit frequency of random surfer
- Probabilistic model
- Stochastic matrices
- Power iteration
- Dead ends (and fix)
- Spider traps (and fix)
- PageRank Equation
  - Extension to topic-specific page-rank
  - Extension to TrustRank

Times Series

- Time series smoothing
  - Moving average
  - Exponential
- Definition of a stationary time series
- Autocorrelation
- AR(p), MA(q), ARMA(p,q) and ARIMA(p,d,q) models
- Hidden Markov Models
  - Relationship of dynamics to random surfer in page rank
  - Relatinoship to mixture models
  - Forward-backward algorithm (see notes)

Social Networks

- Centrality measures
  - Betweenness
  - Closeness
  - Degree
- Girvan-Newman algorithm for clustering
  - Calculating betweenness
  - Selecting number of clusters using the modularity

Social Networks

- **Spectral clustering**
  - Graph cuts
  - Normalized cuts
  - Laplacian Matrix
    - Definition in terms of Adjacency and Degree matrix
    - Properties of eigenvectors
      - Eigenvalues are >= 0
      - First eigenvector
        - Eigenvalue is 0
        - Eigenvector is [1 … 1]^T
      - Second eigenvector (Fiedler vector)
        - Elements sum to 0
        - Eigenvalue is normalized sum of squared edge distances
  - Use of first eigenvector to find normalized cut

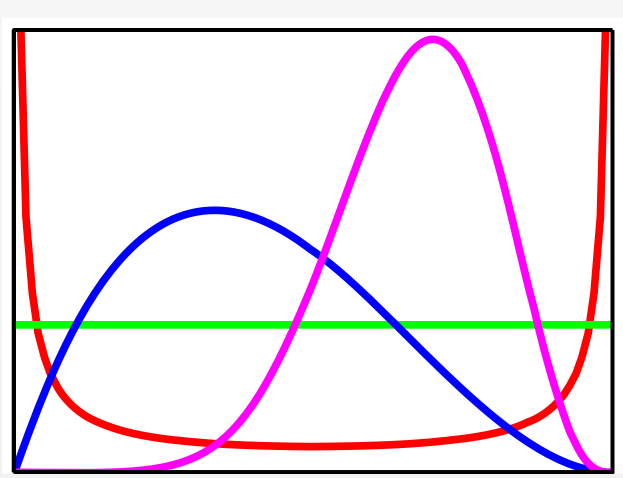
Pre-Midterm Topics

Conjugate Distributions

Binomial: Probability of *m* heads in *N* flips

$$\mathrm{Bin}(m|N,\mu) \quad = \quad \binom{N}{m}\mu^m(1-\mu)^{N-m}$$

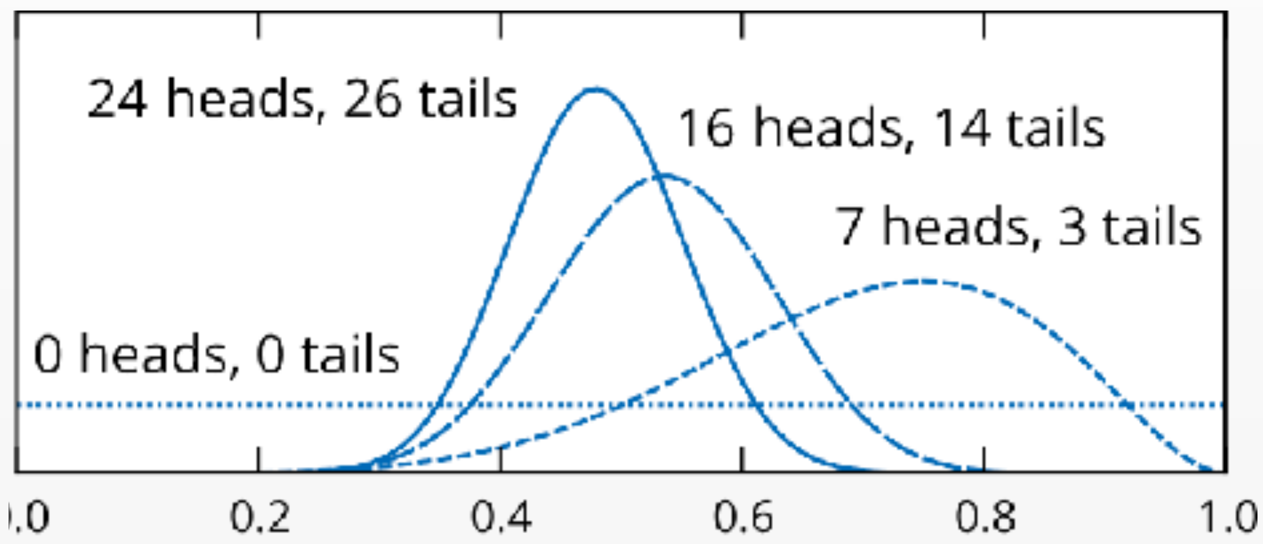
Beta: Probability for bias *μ*



$$\mathrm{Beta}(\mu|a,b) \quad = \quad \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)}\mu^{a-1}(1-\mu)^{b-1}$$

Conjugate Distributions

Posterior probability for μ given flips



$$p(\mu \mid m) = \frac{p(m, \mu)}{p(m)}$$

$$\propto \mathrm{Bin}(m \mid N, \mu)\mathrm{Beta}(\mu \mid a, b)$$

$$\propto \mu^{m+(a-1)}(1-\mu)^{(N-m)+(b-1)}$$

$$p(\mu \mid m) = \mathrm{Beta}(a+m, b+(N-m))$$

Information Theoretic Measures

KL Divergence

$$KL(q\,||\,p) = \sum_x q(x) \log \frac{q(x)}{p(x)}$$

Perplexity

$$\text{Per}(p) = 2^{-\sum_x p(x) \log_2 p(x)}$$

Mutual Information

$$I(X;Y) = \sum_{x,y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)}$$

Perplexity (of a model)

$$\text{Per}(q) = 2^{\sum_{n=1}^{N} \log_2 q(y_n)}$$

$$\hat{p}(y) = \frac{1}{N} \sum_{n=1}^{N} I[y_n = y]$$

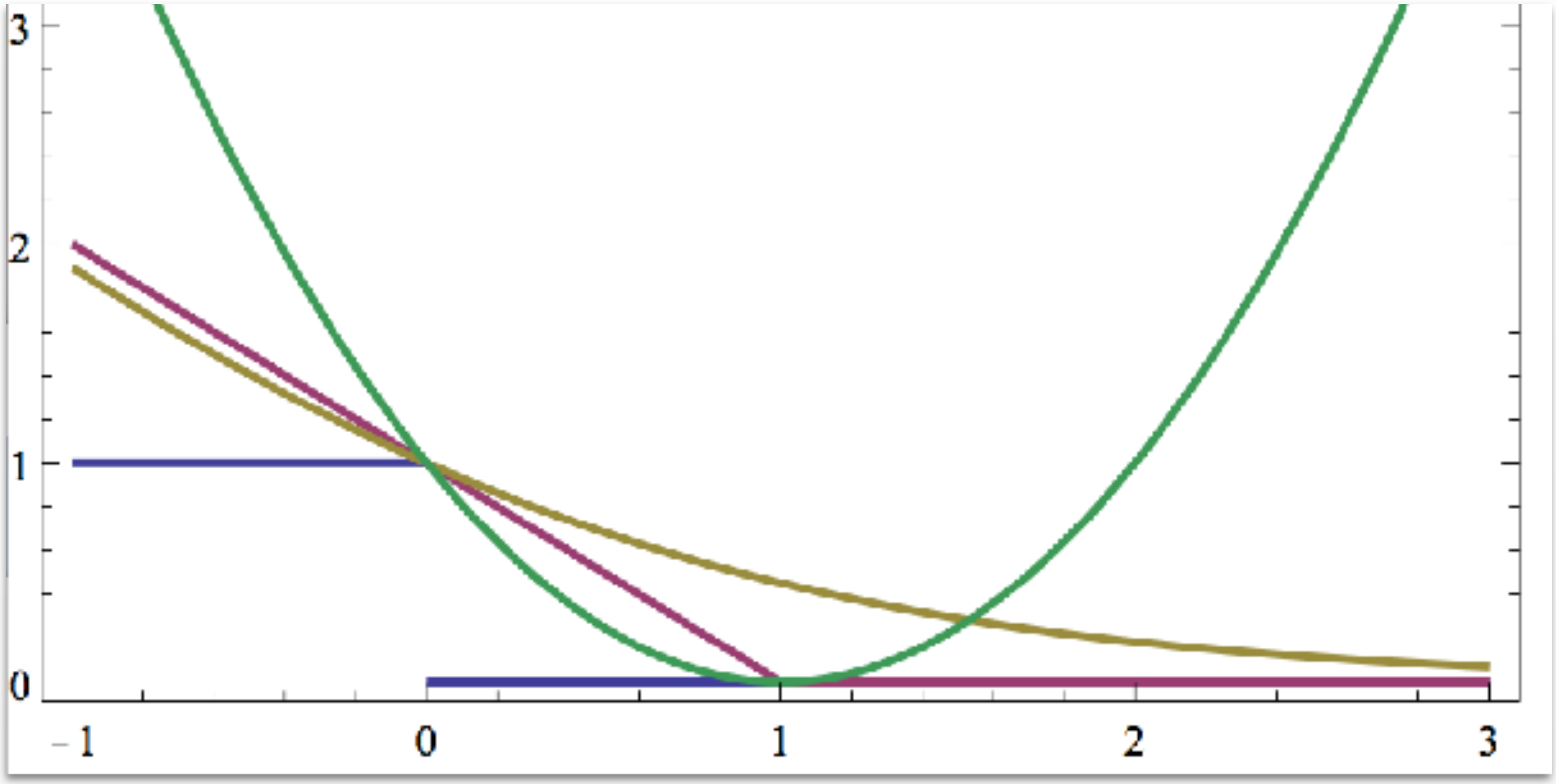
$$H(\hat{p}, q) = -\sum_y \hat{p}(y) \log q(y)$$

$$\text{Per}(q) = e^{H(\hat{p},q)}$$

Entropy

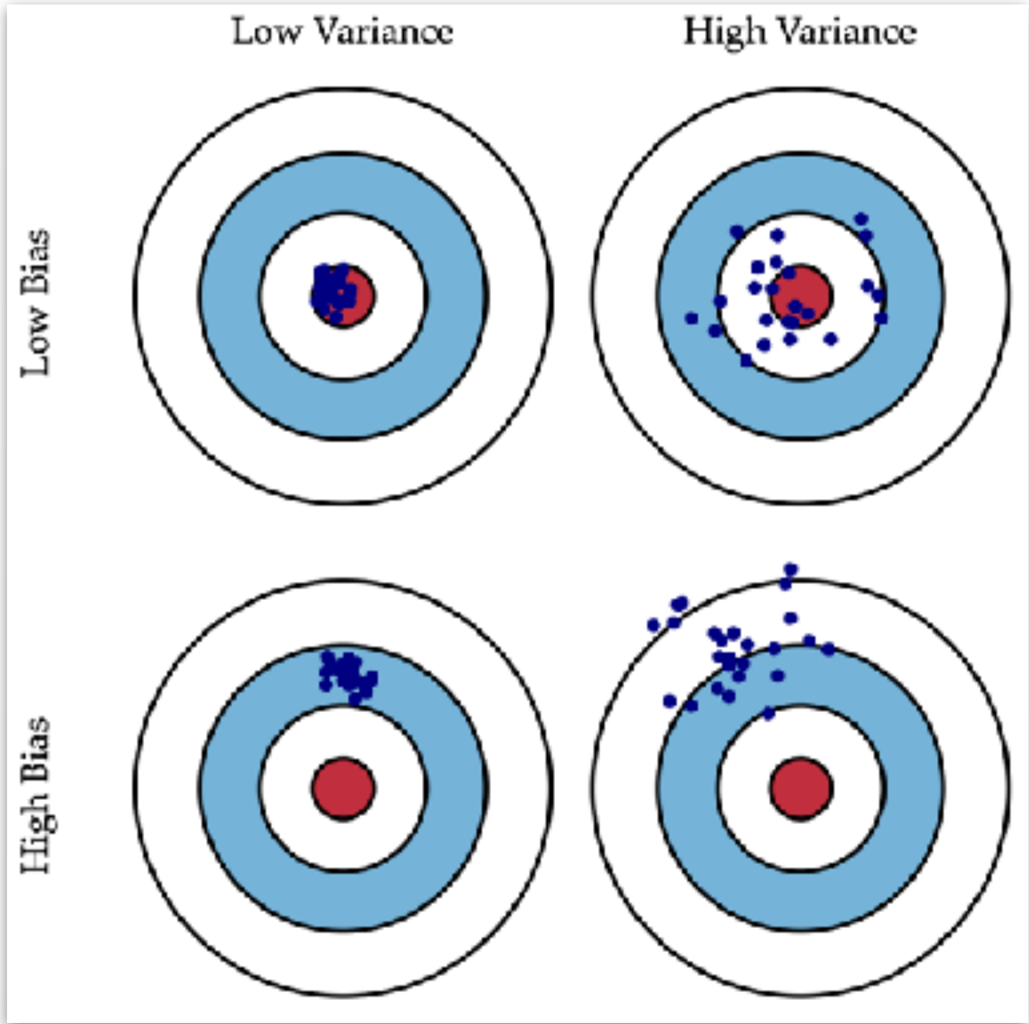
$$H(X) = -\sum_x p(x) \log p(x)$$

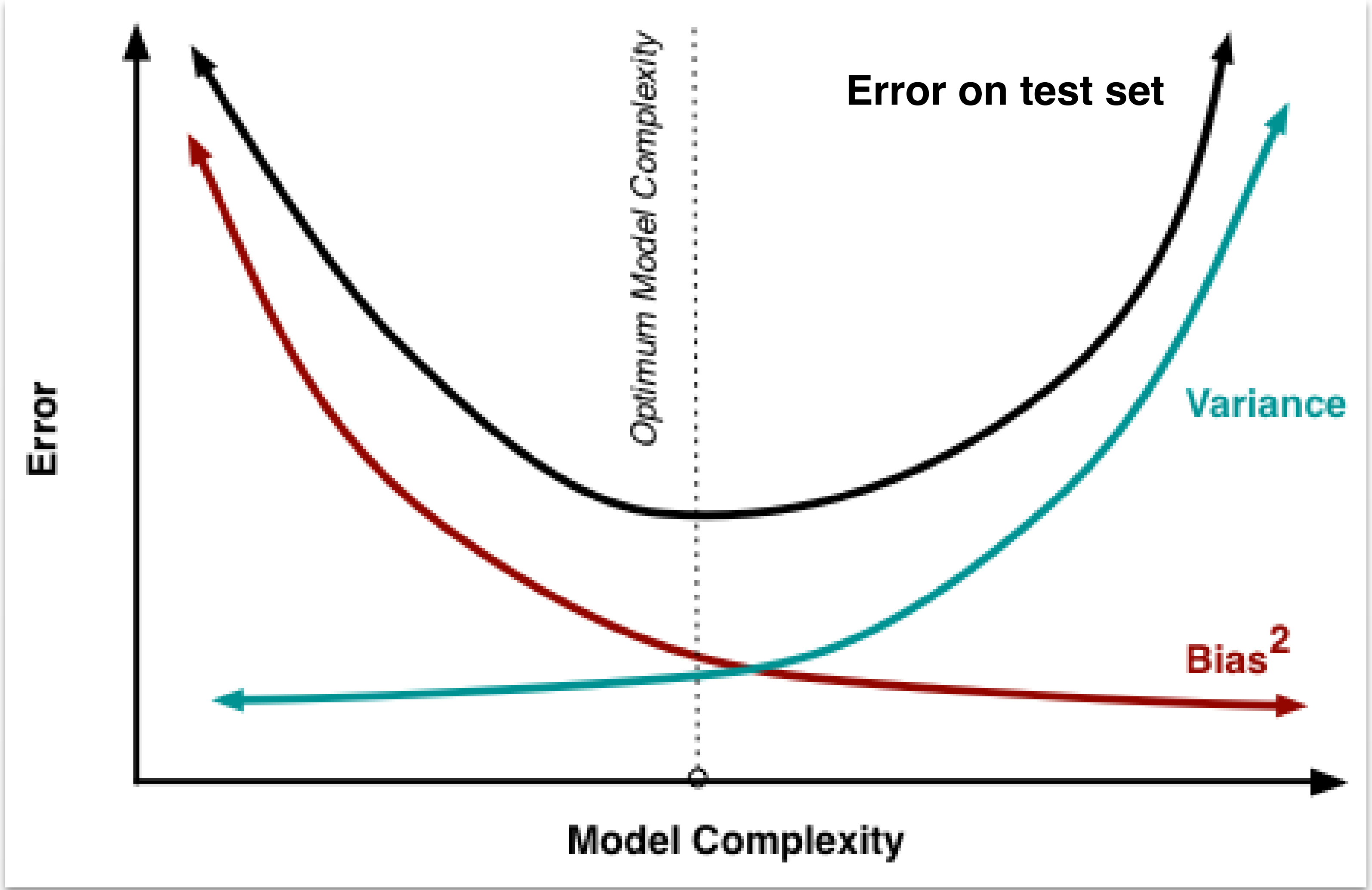
Loss Functions



squared loss:	$\frac{1}{2}(\boldsymbol{w}^\top \boldsymbol{x} - y)^2$	$y \in \mathbb{R}$	Linear Regression
zero-one:	$\frac{1}{4}(\text{Sign}(\boldsymbol{w}^\top \boldsymbol{x}) - y)^2$	$y \in \{-1, +1\}$	Perceptron
logistic loss:	$\log\left(1 + \exp(-y\boldsymbol{w}^\top \boldsymbol{x})\right)$	$y \in \{-1, +1\}$	Logistic Regression
hinge loss:	$\max\{0, 1 - y\boldsymbol{w}^\top \boldsymbol{x}\}$	$y \in \{-1, +1\}$	Soft SVMs

Bias-Variance Trade-Off





Variance of what exactly?

Bias-Variance Trade-Off

Assume classifier predicts expected value for y

$$f(x) = \mathbb{E}_y[y|x] = \bar{y}$$

Squared loss of a classifier

$$\begin{aligned}
\mathbb{E}_y[(y - f(x))^2|x] &= \mathbb{E}_y[(y - \overline{y} + \overline{y} - f(x))^2|x] \\
&= \mathbb{E}_y[(y - \overline{y})^2|x] + \mathbb{E}_y[(\overline{y} - f(x))^2|x] \\
&\quad +2\mathbb{E}_y[(y - \overline{y})(\overline{y} - f(x))|x] \\
&= \mathbb{E}_y[(y - \overline{y})^2|x] + \mathbb{E}_y[(\overline{y} - f(x))^2|x] \\
&\quad +2(\overline{y} - f(x))\mathbb{E}_y[(y - \overline{y})|x] \\
&= \mathbb{E}_y[(y - \overline{y})^2|x] + \mathbb{E}_y[(\overline{y} - f(x))^2|x] \\
&= \mathbb{E}_y[(y - \bar{y})^2|x] + (\bar{y} - f(x))^2
\end{aligned}$$

Bias-Variance Trade-Off

Training Data

$$T = \{(x^i, y^i) | i = 1, \ldots, n\}$$

Classifier/Regressor

$$f_T = \underset{f}{\operatorname{argmin}} \sum_{i=1}^{N} \mathcal{L}(y_i, f(x^i))$$

Expected value for y

$$\bar{y} = \mathbb{E}_y[y|x]$$

Expected prediction

$$\bar{f}(x) = \mathbb{E}_T[f_T(x)]$$

Bias-Variance Decomposition

$$\begin{aligned}
\mathbb{E}_{y,T}[(y - f_T(x))^2 | x] = \mathbb{E}_y[(y - \bar{y})^2 | x] \\
+ \mathbb{E}_{y,T}[(\bar{f}(x) - f_T(x))^2 | x] \\
+ \mathbb{E}_y[(\bar{y} - \bar{f}(x))^2 | x] \\
= \operatorname{var}_y(y|x) + \operatorname{var}_T(f(x)) + \operatorname{bias}(f_T(x))^2
\end{aligned}$$

Bagging and Boosting

Bagging

$$F_T^{\text{bag}}(x) = \frac{1}{B}\sum_{b=1}^{B} f_{T_b}(x)$$

- Sample B datasets T_b at random with replacement from the full data T
- Train on classifiers independently on each dataset and average results
- Decreases variance (i.e. overfitting) does not affect bias (i.e. accuracy).

Boosting

$$F^{\text{boost}}(x) = \frac{1}{B}\sum_{b=1}^{B} \alpha_b f_{w_b}(x)$$

- Sequential training
- Assign higher weight to previously misclassified data points
- Combines weighted weak learners (high bias) into a strong learner (low bias)
- Also some reduction of variance (in later iterations)