An Introduction to Ensemble and Boosting Methods

Amir Saffari

Institute for Computer Graphics and Vision (ICG)
Graz University of Technology, Austria
http://www.ymer.org/amir/
saffari@icg.tugraz.at , amir@ymer.org

PASCAL Bootcamp 2007
Vilanova i la Geltrú, Spain

Outline

Outline

Ensemble Methods

Introduction

Model Averaging

Bagging

Stagewise Additive Modeling

Stagewise Additive Modeling

Boosting

Practical Example

Choosing your operating system

Majority voting scheme

- Ask experts for their opinion and choose the option with majority vote.
- Let's say we have a set of M experts:
  $H = \{f_1, f_2, \ldots, f_M\}, \quad f_m(budget) \in \{Linux, Windows\}$
- Assume $Linux = +1$, $Windows = -1$, then the majority vote decision will be:
  $F(budget) = sign(\frac{1}{M}\sum_{m=1}^{M} f_m(budget))$
- This is the main concept behind ensemble methods.
- Diversity is just more than great.

Majority voting scheme

- Ask experts for their opinion and choose the option with majority vote.
- Let's say we have a set of M experts: $H = \{f_1, f_2, \ldots, f_M\}, \quad f_m(budget) \in \{Linux, Windows\}$
- Assume $Linux = +1$, $Windows = -1$, then the majority vote decision will be: $F(budget) = sign(\frac{1}{M}\sum_{m=1}^{M} f_m(budget))$
- This is the main concept behind ensemble methods.
- Diversity is just more than great.

Majority voting scheme

- Ask experts for their opinion and choose the option with majority vote.
- Let's say we have a set of M experts:
  $H = \{f_1, f_2, \ldots, f_M\}, \quad f_m(budget) \in \{Linux, Windows\}$
- Assume $Linux = +1$, $Windows = -1$, then the majority vote decision will be:
  $F(budget) = sign(\frac{1}{M}\sum_{m=1}^{M} f_m(budget))$
- This is the main concept behind ensemble methods.
- Diversity is just more than great.

Majority voting scheme

- Ask experts for their opinion and choose the option with majority vote.
- Let's say we have a set of M experts:
  $H = \{f_1, f_2, \dots, f_M\}, \quad f_m(budget) \in \{Linux, Windows\}$
- Assume $Linux = +1$, $Windows = -1$, then the majority vote decision will be:
  $F(budget) = sign(\frac{1}{M}\sum_{m=1}^{M} f_m(budget))$
- This is the main concept behind ensemble methods.
- Diversity is just more than great.

Majority voting scheme

- Ask experts for their opinion and choose the option with majority vote.
- Let's say we have a set of M experts:
  $H = \{f_1, f_2, \ldots, f_M\}, \quad f_m(budget) \in \{Linux, Windows\}$
- Assume $Linux = +1$, $Windows = -1$, then the majority vote decision will be:
  $F(budget) = sign(\frac{1}{M}\sum_{m=1}^{M} f_m(budget))$
- This is the main concept behind ensemble methods.
- Diversity is just more than great.

Notations

- $D = \{(\mathbf{x}_1, t_1), (\mathbf{x}_2, t_2), \ldots, (\mathbf{x}_N, t_N)\}$
- $\mathbf{x}_n \in R^d, t_n \in \{-1, +1\}$
- $H = \{f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_M(\mathbf{x})\}$
- $y_m = f_m(\mathbf{x}) \in \{-1, +1\}$
- $F(\mathbf{x}) = \sum_{m=1}^{M} \alpha_m f_m(\mathbf{x})$
- $\alpha_m \in R^+, \sum_{m=1}^{M} \alpha_m = 1$

Outline

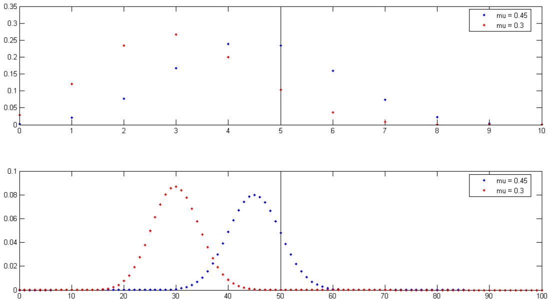
Why to use ensemble methods?

Better performance

Assume that: $\forall j : p(y_m \neq t) \leq \mu < 1/2$, and the decisions of different models are independent, then the chance of a wrong decision by the ensemble, $p(F \neq t) = 1 - Pr(k \leq M/2)$, where $Pr(k \leq K)$ is the cumulative distribution function of a binomial distribution.

This upper bound is pretty much better than the original error rate.

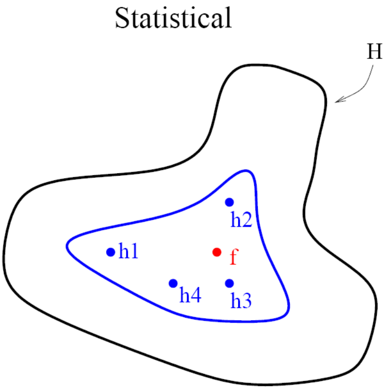
Performance of ensemble of classifiers



For $\mu = 0.3$ and $M = 21$, the chance of misclassification is around 0.026 (T. G. Diettrich 2000).

Why to use ensemble methods?

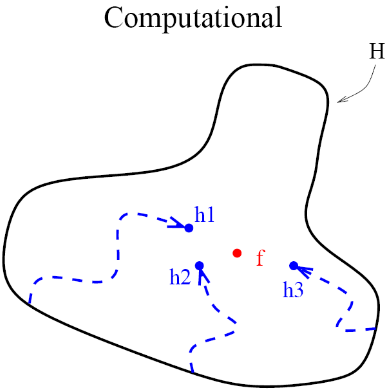
Statistical reason



From: T. G. Diettrich, Ensemble Methods in Machine Learning, Lecture Notes in Computer Science, Vol. 1857, pages: 1-15, 2000.

Why to use ensemble methods?

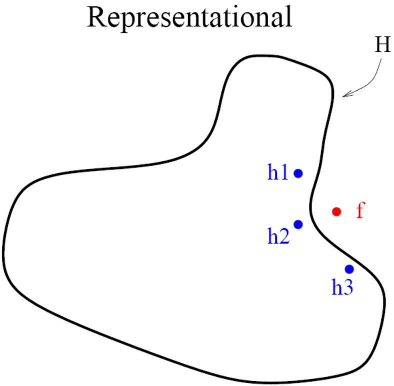
Computational reason



From: T. G. Diettrich, Ensemble Methods in Machine Learning, Lecture Notes in Computer Science, Vol. 1857, pages: 1-15, 2000.

Why to use ensemble methods?

Representational reason



From: T. G. Diettrich, Ensemble Methods in Machine Learning, Lecture Notes in Computer Science, Vol. 1857, pages: 1-15, 2000.

Why to use ensemble methods?

- Computational efficiency We are looking for a set of weak learners (classifiers, or hypotheses): $p(y \neq t) < 1/2$.
- Different classes of base models Choices could be: Trees (stumps, small, large), Naive Bayes, k-Nearest Neighbors, Neural Networks, Linear SVM, YOUR-MAGICAL-MODEL, ...

Why to use ensemble methods?

- Computational efficiency We are looking for a set of weak learners (classifiers, or hypotheses): $p(y \neq t) < 1/2$.
- Different classes of base models Choices could be: Trees (stumps, small, large), Naive Bayes, k-Nearest Neighbors, Neural Networks, Linear SVM, YOUR-MAGICAL-MODEL, ...

How to find the base models?

- Train a diverse set of models on the same datasets.
- Train a set of models from a specific class of learners by using diversity in the datasets, parameters, or initial conditions.
- Cross-validated committees
- Bagging
- Boosting

How to find the base models?

- Train a diverse set of models on the same datasets.
- Train a set of models from a specific class of learners by using diversity in the datasets, parameters, or initial conditions.
- Cross-validated committees
- Bagging
- Boosting

How to find the base models?

- Train a diverse set of models on the same datasets.
- Train a set of models from a specific class of learners by using diversity in the datasets, parameters, or initial conditions.
- Cross-validated committees
- Bagging
- Boosting

How to find the base models?

- Train a diverse set of models on the same datasets.
- Train a set of models from a specific class of learners by using diversity in the datasets, parameters, or initial conditions.
- Cross-validated committees
- Bagging
- Boosting

Outline

Bagging

- Create subsets of the training samples, called bootstrap replicates, each containing examples drawn randomly with replacement from the original training dataset, and train learning algorithms over them.
- The method is called bootstrap aggregation.
- Originally developed to reduce the variance of the learning algorithms.

L. Breiman, Bagging Predictors, Machine Learning, Vol. 24, pages: 123-140, 1996.

Bagging

- Create subsets of the training samples, called bootstrap replicates, each containing examples drawn randomly with replacement from the original training dataset, and train learning algorithms over them.
- The method is called bootstrap aggregation.
- Originally developed to reduce the variance of the learning algorithms.

L. Breiman, Bagging Predictors, Machine Learning, Vol. 24, pages: 123-140, 1996.

Bagging

- Create subsets of the training samples, called bootstrap replicates, each containing examples drawn randomly with replacement from the original training dataset, and train learning algorithms over them.
- The method is called bootstrap aggregation.
- Originally developed to reduce the variance of the learning algorithms.

L. Breiman, Bagging Predictors, Machine Learning, Vol. 24, pages: 123-140, 1996.

Outline

Stagewise additive modeling

$F(\mathbf{x}) = \sum_{m=1}^{M} \alpha_m f_m(\mathbf{x})$

General Forward Stagewise Additive Modeling

- Set $F^{(0)}(\mathbf{x}) = 0$
- for m = 1 to M, do
- $\{f_m(\mathbf{x}), \alpha_m\} = \underset{f,\alpha}{argmin} \sum_{n=1}^{N} L(t_n, F^{(m-1)}(\mathbf{x}_n) + \alpha f(\mathbf{x}_n))$
- $F^{(m)}(\mathbf{x}) = F^{(m-1)}(\mathbf{x}) + \alpha_m f_m(\mathbf{x})$

J. Friedman, T. Hastie, R. Tibshirani, Additive Logistic Regression: a Statistical View of Boosting, Annals of Statistics, Vol. 28, pages: 337-407, 2000.

Stagewise additive modeling

$F(\mathbf{x}) = \sum_{m=1}^{M} \alpha_m f_m(\mathbf{x})$

General Forward Stagewise Additive Modeling

- Set $F^{(0)}(\mathbf{x}) = 0$
- for m = 1 to M, do
- $\{f_m(\mathbf{x}), \alpha_m\} = \underset{f,\alpha}{argmin} \sum_{n=1}^{N} L(t_n, F^{(m-1)}(\mathbf{x}_n) + \alpha f(\mathbf{x}_n))$
- $F^{(m)}(\mathbf{x}) = F^{(m-1)}(\mathbf{x}) + \alpha_m f_m(\mathbf{x})$

J. Friedman, T. Hastie, R. Tibshirani, Additive Logistic Regression: a Statistical View of Boosting, Annals of Statistics, Vol. 28, pages: 337-407, 2000.

Stagewise additive modeling

$F(\mathbf{x}) = \sum_{m=1}^{M} \alpha_m f_m(\mathbf{x})$

General Forward Stagewise Additive Modeling

- Set $F^{(0)}(\mathbf{x}) = 0$
- for m = 1 to M, do
- $\{f_m(\mathbf{x}), \alpha_m\} = \underset{f,\alpha}{argmin} \sum_{n=1}^{N} L(t_n, F^{(m-1)}(\mathbf{x}_n) + \alpha f(\mathbf{x}_n))$
- $F^{(m)}(\mathbf{x}) = F^{(m-1)}(\mathbf{x}) + \alpha_m f_m(\mathbf{x})$

J. Friedman, T. Hastie, R. Tibshirani, Additive Logistic Regression: a Statistical View of Boosting, Annals of Statistics, Vol. 28, pages: 337-407, 2000.

Stagewise additive modeling

$F(\mathbf{x}) = \sum_{m=1}^{M} \alpha_m f_m(\mathbf{x})$

General Forward Stagewise Additive Modeling

- Set $F^{(0)}(\mathbf{x}) = 0$
- for m = 1 to M, do
- $\{f_m(\mathbf{x}), \alpha_m\} = \underset{f,\alpha}{argmin} \sum_{n=1}^{N} L(t_n, F^{(m-1)}(\mathbf{x}_n) + \alpha f(\mathbf{x}_n))$
- $F^{(m)}(\mathbf{x}) = F^{(m-1)}(\mathbf{x}) + \alpha_m f_m(\mathbf{x})$

J. Friedman, T. Hastie, R. Tibshirani, Additive Logistic Regression: a Statistical View of Boosting, Annals of Statistics, Vol. 28, pages: 337-407, 2000.

Outline

AdaBoost

$F(\mathbf{x}) = \sum_{m=1}^{M} \alpha_m f_m(\mathbf{x})$
$l(t, y) = -t.y$

Discrete AdaBoost

- Set $W = \{w_1, w_2, \ldots, w_N\}, \forall n : w_n = 1/N$
- for m = 1 to M, do
- $f_m(\mathbf{x}) = \underset{f}{argmin} \sum_{n=1}^{N} w_n (t_n - f(\mathbf{x}_n))^2$
- $e_m = \sum_{n=1}^{N} w_n l(t_n, f_m(\mathbf{x}_n))$
- $\alpha_m = \log \frac{1-e_m}{e_m}$
- $w_n \leftarrow w_n \exp(\alpha_m l(t_n, f_m(\mathbf{x}_n)))$
- $w_n \leftarrow \sum_{n=1}^{N} w_n$

Y. Freund, R. Schapire, Experiments with a New Boosting Algorithm, Proceedings of ICML, pages: 148-156, 1997.

AdaBoost

$F(\mathbf{x}) = \sum_{m=1}^{M} \alpha_m f_m(\mathbf{x})$
$l(t, y) = -t.y$

Discrete AdaBoost

- Set $W = \{w_1, w_2, \ldots, w_N\}, \forall n : w_n = 1/N$
- for m = 1 to M, do
- $f_m(\mathbf{x}) = \underset{f}{argmin} \sum_{n=1}^{N} w_n(t_n - f(\mathbf{x}_n))^2$
- $e_m = \sum_{n=1}^{N} w_n l(t_n, f_m(\mathbf{x}_n))$
- $\alpha_m = \log \frac{1-e_m}{e_m}$
- $w_n \leftarrow w_n \exp(\alpha_m l(t_n, f_m(\mathbf{x}_n)))$
- $w_n \leftarrow \sum_{n=1}^{N} w_n$

Y. Freund, R. Schapire, Experiments with a New Boosting Algorithm, Proceedings of ICML, pages: 148-156, 1997.

AdaBoost

$F(\mathbf{x}) = \sum_{m=1}^{M} \alpha_m f_m(\mathbf{x})$
$l(t, y) = -t.y$

Discrete AdaBoost

- Set $W = \{w_1, w_2, \ldots, w_N\}, \forall n : w_n = 1/N$
- for m = 1 to M, do
- $f_m(\mathbf{x}) = \underset{f}{argmin} \sum_{n=1}^{N} w_n(t_n - f(\mathbf{x}_n))^2$
- $e_m = \sum_{n=1}^{N} w_n l(t_n, f_m(\mathbf{x}_n))$
- $\alpha_m = \log \frac{1-e_m}{e_m}$
- $w_n \leftarrow w_n \exp(\alpha_m l(t_n, f_m(\mathbf{x}_n)))$
- $w_n \leftarrow \sum_{n=1}^{N} w_n$

Y. Freund, R. Schapire, Experiments with a New Boosting Algorithm, Proceedings of ICML, pages: 148-156, 1997.

AdaBoost

$F(\mathbf{x}) = \sum_{m=1}^{M} \alpha_m f_m(\mathbf{x})$
$l(t, y) = -t.y$

Discrete AdaBoost

- Set $W = \{w_1, w_2, \ldots, w_N\}, \forall n : w_n = 1/N$
- for m = 1 to M, do
- $f_m(\mathbf{x}) = \underset{f}{argmin} \sum_{n=1}^{N} w_n(t_n - f(\mathbf{x}_n))^2$
- $e_m = \sum_{n=1}^{N} w_n l(t_n, f_m(\mathbf{x}_n))$
- $\alpha_m = \log \frac{1-e_m}{e_m}$
- $w_n \leftarrow w_n \exp(\alpha_m l(t_n, f_m(\mathbf{x}_n)))$
- $w_n \leftarrow \sum_{n=1}^{N} w_n$

Y. Freund, R. Schapire, Experiments with a New Boosting Algorithm, Proceedings of ICML, pages: 148-156, 1997.

AdaBoost

$F(\mathbf{x}) = \sum_{m=1}^{M} \alpha_m f_m(\mathbf{x})$
$l(t, y) = -t.y$

Discrete AdaBoost

- Set $W = \{w_1, w_2, \ldots, w_N\}, \forall n : w_n = 1/N$
- for m = 1 to M, do
- $f_m(\mathbf{x}) = \underset{f}{argmin} \sum_{n=1}^{N} w_n (t_n - f(\mathbf{x}_n))^2$
- $e_m = \sum_{n=1}^{N} w_n l(t_n, f_m(\mathbf{x}_n))$
- $\alpha_m = \log \frac{1-e_m}{e_m}$
- $w_n \leftarrow w_n \exp(\alpha_m l(t_n, f_m(\mathbf{x}_n)))$
- $w_n \leftarrow \sum_{n=1}^{N} w_n$

Y. Freund, R. Schapire, Experiments with a New Boosting Algorithm, Proceedings of ICML, pages: 148-156, 1997.

AdaBoost

$F(\mathbf{x}) = \sum_{m=1}^{M} \alpha_m f_m(\mathbf{x})$
$l(t, y) = -t.y$

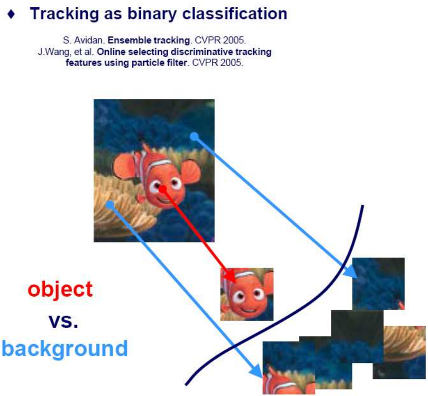
Discrete AdaBoost

- Set $W = \{w_1, w_2, \ldots, w_N\}, \forall n : w_n = 1/N$
- for m = 1 to M, do
- $f_m(\mathbf{x}) = \underset{f}{argmin} \sum_{n=1}^{N} w_n (t_n - f(\mathbf{x}_n))^2$
- $e_m = \sum_{n=1}^{N} w_n l(t_n, f_m(\mathbf{x}_n))$
- $\alpha_m = \log \frac{1-e_m}{e_m}$
- $w_n \leftarrow w_n \exp(\alpha_m l(t_n, f_m(\mathbf{x}_n)))$
- $w_n \leftarrow \sum_{n=1}^{N} w_n$

Y. Freund, R. Schapire, Experiments with a New Boosting Algorithm, Proceedings of ICML, pages: 148-156, 1997.

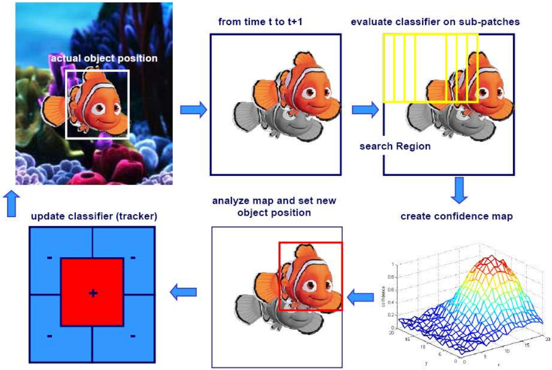
Outline

Tracking visual objects



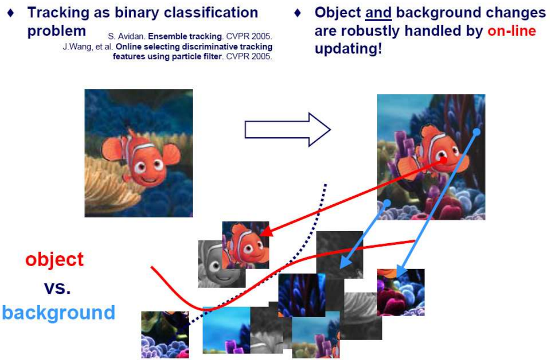
H. Grabner, M. Grabner, H. Bischof, Real-Time Tracking via On-line Boosting, BMVC, 2006.

Tracking visual objects



H. Grabner, M. Grabner, H. Bischof, Real-Time Tracking via On-line Boosting, BMVC, 2006.

Tracking visual objects



H. Grabner, M. Grabner, H. Bischof, Real-Time Tracking via On-line Boosting, BMVC, 2006.