

Alpaydin's Introduction to Machine Learning

- **Publisher:** The MIT Press
- **Author:** Ethem Alpaydin
- **Presenter:** Wen-Bin Luo
- **Link:** <https://mitpress.mit.edu/books/introduction-machine-learning-third-edition>

Contents

- Introduction
- Supervised Learning
- Bayesian Decision Theory
- Parametric Methods
- Multivariate Methods
- Dimensionality Reduction

Supervised Learning

- Learning a Class from Examples
- Vapnik-Chervonenkis Dimension
- Probably Approximately Correct Learning
- Noise
- Learning Multiple Classes
- Regression
- Model Selection and Generalization
- Dimensions of a Supervised Machine Learning Algorithm
- Notes

Vapnik-Chervonenkis Dimension

- Given a dataset containing N points, a hypothesis H shatters N points if it separates the positive examples from the negative.
- **Vapnik-Chervonenkis (VC) dimension** of a hypothesis H , denoted as $VC(H)$: The maximum number of points that can be shattered by H .

Probably Approximately Correct Learning

- Given a class C , and examples drawn from some unknown but fixed probability distribution $p(x)$.
- We want to find the number of examples, N , such that with probability at least $1 - \delta$, the hypothesis H has error at most ε , for arbitrary $\delta \leq \frac{1}{2}$ and $\varepsilon > 0$.
- $P(C\Delta H \leq \varepsilon) \geq 1 - \delta$, where $C\Delta H$ is the region of difference between C and H .

Noise

- Noise can result from:
 - Imprecision in recording the input attributes.
 - Errors in labeling the data points.
 - Hidden or latent attributes that may be unobservable.
- **Occam's razor**: simpler explanations are more plausible and any unnecessary complexity should be shaved off.

Regression

- If there is no noise, the task is *interpolation/extrapolation*.
- In regression, there is noise added to the output of the unknown function.

Model Selection and Generalization

- An **ill-posed problem** is where the data by itself is not sufficient to find a unique solution.
- **Model selection**: Choosing between possible hypothesis.
- **Generalization**: How well a model trained on the training set predicts the right output for new instances.
- **Triple trade-off**:
 - The complexity of the hypothesis we fit to data.
 - The amount of training data.
 - The generalization error on new examples.
- In general, as the complexity of a model class increases, the generalization error decreases first and then starts to increase.
- Datasets:
 - **Training set**: To train the model.
 - **Validation set**: To test the generalization ability.

- **Test set**, or **publication set**: To report the error to give an idea about the expected error of our best model.
- In *cross-validation*, the hypothesis that is the most accurate on the validation set is the best one.

Dimensions of a Supervised Machine Learning Algorithm

- There are three decisions we must make:
 - **Model**: Denoted as $\hat{f}(x|\theta)$ where \hat{f} is the model, x is the input, and θ are the parameters.
 - **Loss function** (L): To compute the difference between the desired output and our approximation to it.
 - **Optimization procedure**: To find θ^* that minimizes the total error.

Bayesian Decision Theory

- Introduction
- Classification
- Losses and Risks
- Discriminant Functions
- Association Rules
- Notes

Classification

- **Bayes' rule**: $P(y = i|x) = P(y = i)P(x|y = i)/P(x)$ where
 - $P(y = i|x)$ is the **posterior probability**.
 - $P(y = i)$ is the **prior probability**.
 - $P(x|y = i)$ is the **likelihood**.
 - $P(x)$ is the **evidence**.
- **Bayes' classifier**: Given an observation x , the predicted class $\hat{y} = \operatorname{argmax}_i P(y = i|x)$.

Losses and Risks

- Let λ_{ik} be the loss incurred for falsely assuming $\hat{y} = i$ when the input actually belongs to $y = k$.
- The *expected loss* for misclassification is $L(y = i|x) = \sum_{k=1}^K \lambda_{ik}P(y = k|x)$.
- The class with the least expected loss is $\operatorname{argmin}_i L(y = i|x)$.
- In Bayesian classifier, λ_{ik} is 0 if $i = k$, or 1 if $i \neq k$.

- $\hat{y} = \operatorname{argmin}_i L(y = i|x) = \operatorname{argmin}_i \sum_{k=1}^K \lambda_{ik} P(y = k|x) = \operatorname{argmin}_i 1 - P(y = i|x) = \operatorname{argmax}_i P(y = i|x)$.

Discriminant Functions

- Classification can be seen as implementing a set of *discriminant functions*, $g_i(x)$, $i \in \{1, \dots, K\}$, such that $\hat{y} = \operatorname{argmax}_i g_i(x)$.
- This divides the feature space into K *decision regions* R_i , $i \in \{1, \dots, K\}$.
- The regions are separated by *decision boundaries*.

Association Rules

- An association rule is an implication of the form $X \rightarrow Y$ where X is the **antecedent** and Y is the **consequent** of the rule.
- **Support**: $\operatorname{support}(X \rightarrow Y) := P(X, Y)$.
- **Confidence**: $\operatorname{confidence}(X \rightarrow Y) := P(Y|X)$.
- **Lift** (or **interest**): $\operatorname{lift}(X \rightarrow Y) := \frac{P(X,Y)}{P(X)P(Y)} = \frac{P(Y|X)}{P(Y)}$.
- Two steps of **Apriori** algorithm:
 1. Find frequent item sets, that is, those which have enough *support*.
 2. Convert them to rules with enough *confidence* by splitting the items into two, as items in the *antecedent* and items in the *consequent*.
- A rule $X \rightarrow Y$ need not imply causality but just an association.
- In a problem, there may also be *hidden variables* whose values are never known through evidence.

Parametric Methods

- Introduction
- Maximum Likelihood Estimation
- Evaluating an Estimator: Bias and Variance
- The Bayes' Estimator
- Parametric Classification
- Regression
- Tuning Model Complexity: Bias/Variance Dilemma
- Model Selection Procedures

Maximum Likelihood Estimation

- Let $X = \{x_i\}_{i=1}^N$ be a set of N independent and identically distributed (iid) samples drawn from some known probability density family.
- The **likelihood** of parameter θ given sample X is the product of the likelihoods of the individual points: $I(\theta|X) = P(X|\theta) = \prod_{i=1}^N P(x_i|\theta)$.
- **Log likelihood**: $L(\theta|X) = \log I(\theta|X) = \log P(X|\theta) = \sum_{i=1}^N \log P(x_i|\theta)$.
- **Maximum likelihood estimation (MLE)**: $\hat{\theta} = \operatorname{argmax}_{\theta} I(\theta|X) = \operatorname{argmax}_{\theta} L(\theta|X)$.
- **Bernoulli density**:
 - $X \sim B(N, \theta)$.
 - $P(x_i|\theta) = \theta^{x_i}(1 - \theta)^{1-x_i}$.
 - $L(\theta|X) = \log \prod_{i=1}^N \theta^{x_i}(1 - \theta)^{1-x_i} = \sum_i x_i \log \theta + (N - \sum_i x_i) \log(1 - \theta)$.
 - $\hat{\theta} = \sum_i x_i / N$.
- **Multinomial density**:
 - $X \sim \text{multinomial}(N, \theta)$, where $\theta = \{\theta_i | i = 1, \dots, K\}$.
 - $P(x_i|\theta) = \prod_{k=1}^K \theta_k^{x_{ik}}$ where x_{ik} is 1 if $x_i = k$, or 0 if $x_i \neq k$.
 - $\hat{\theta}_k = \sum_i x_{ik} / N$, $k \in \{1, \dots, K\}$.
- **Gaussian density**:
 - $X \sim N(\mu, \sigma^2)$.
 - $P(x_i|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x_i - \mu)^2}{2\sigma^2}\right)$.
 - $\hat{\mu} = \sum_i x_i / N$.
 - $\hat{\sigma}^2 = \sum_i (x_i - \hat{\mu})^2 / N$.

Evaluating an Estimator: Bias and Variance

- Let $\hat{\theta}$ be an estimator of θ based on N observations.
- **Bias** of an estimator: $b_{\theta}(\hat{\theta}) := E[\theta - \hat{\theta}]$.
- **Mean square error (MSE)** of an estimator: $r_{\theta}(\hat{\theta}) := E[(\theta - \hat{\theta})^2]$.
- **Unbiased estimator**: $\hat{\theta}$ is an *unbiased* estimator of θ if $b_{\theta}(\hat{\theta}) = 0$ or $E[\hat{\theta}] = \theta$.
- **Consistent estimator**: $\hat{\theta}$ is a *consistent* estimator of θ if $r_{\theta}(\hat{\theta}) \rightarrow 0$ as $N \rightarrow \infty$.
- $m = \sum_i x_i / N$ is an unbiased and consistent estimator of μ .
- $s^2 = \sum (x_i - m)^2 / N$ is a biased but consistent estimator of σ^2 since $E[s^2] = \frac{N-1}{N} \sigma^2 \neq \sigma^2$.
- **Asymptotically unbiased estimator**: $\hat{\theta}$ is an *asymptotically unbiased* estimator of θ if $b_{\theta}(\hat{\theta}) \rightarrow 0$ or $E[\hat{\theta}] \rightarrow \theta$ as $N \rightarrow \infty$.
- $\text{MSE} = r_{\theta}(\hat{\theta}) = b_{\theta}^2(\hat{\theta}) + \text{variance}(\hat{\theta}) = \text{bias}^2 + \text{variance}$.

The Bayes' Estimator

- The estimation of θ can be exploited by prior information on the distribution of θ .
- **Bayes' rule:** $P(\theta|X) = P(\theta)P(X|\theta)/P(X)$ where
 - **Posterior density** $P(\theta|X)$: the likely θ values after looking at the sample.
 - **Prior density** $P(\theta)$: the likely values that θ may take before looking at the sample.
- **Maximum likelihood estimate (MLE):** $\hat{\theta} = \operatorname{argmax}_{\theta} P(X|\theta)$.
- **Maximum a posteriori (MAP) estimate:** $\hat{\theta} = \operatorname{argmax}_{\theta} P(\theta|X)$.
- **Bayes' estimate:** $\hat{\theta} = E[\theta|X] = \int \theta P(\theta|X) d\theta$.
- The Bayes' estimator for posterior mean $\hat{\mu}$ is a weighted average of the prior mean μ and the sample mean m .

Parametric Classification

- In *Bayes' classification*, the discriminant function for class $i \in \{1, \dots, K\}$ is
 - $g_i(x) = P(x|y=i)P(y=i)$
 - $g_i(x) = \log P(x|y=i) + \log P(y=i)$.
- **Gaussian Bayes' classification:**
 - Assume $P(x|y=i) \sim N(\mu_i, \sigma_i^2)$.
 - $g_i(x) = -\frac{1}{2} \log 2\pi - \log \sigma_i - \frac{(x-\mu_i)^2}{2\sigma_i^2} + \log P(y=i)$.
 - $\mu_i \sim m_i$ and $\sigma_i^2 \sim s_i^2$ are estimated from N observations using maximum likelihood estimation.
- *Simplified Gaussian Bayes' classification:*
 - Assumption(s):
 - Equal variances, i.e., $\sigma^2 = \sigma_i^2$ for class $i \in \{1, \dots, K\}$.
 - Equal priors, i.e., $P = P(y=i)$ for class $i \in \{1, \dots, K\}$.
 - $g_i(x) \propto -\frac{(x-m_i)^2}{2s_i^2}$ and $\operatorname{argmax}_i g_i(x) = \operatorname{argmin}_i |x - m_i|$ if
 - The decision boundary is the midpoint between the two means.

Regression

- $y = f(x) + \varepsilon$: The numeric output is the sum of a deterministic function of the input and random noise.
- $f(x)$, the unknown function, is approximated by the estimator $\hat{f}(x|\theta)$.
- Assume that ε is zero mean Gaussian with constant variance σ^2 , namely, $\varepsilon \sim N(0, \sigma^2)$.
- By placing $\hat{f}(x|\theta)$ in place of $f(x)$, we have $P(y|x) \sim N(\hat{f}(x|\theta), \sigma^2)$.
- $P(x, y) = P(y|x)P(x)$, where $P(y|x)$ is the output given the input, and $P(x)$ is the input density.
- $L(\theta|X) = \log \prod_{i=1}^N P(x_i, y_i) = \log \prod_{i=1}^N P(y_i|x_i) + \log \prod_{i=1}^N P(x_i)$.
- **Linear regression:**

- Assume Gaussian distributed error.
- Maximizing likelihood corresponds to minimizing the sum of squared errors.
- $\hat{f}(x|w_0, w_1) = w_0 + w_1x$.
- $\operatorname{argmax}_{w_0, w_1} L(w_0, w_1|X) = \operatorname{argmax}_{w_0, w_1} \log \prod_{i=1}^N P(y_i|x_i) = \operatorname{argmax}_{w_0, w_1} \sum_{i=1}^N (y_i - \hat{f}(x_i|w_0, w_1))^2$.
- **Relative squared error (RSE):** RSE = residual sum of squares (RSS) / total sum of squares (TSS) = $\sum_i (y_i - \hat{y}_i)^2 / \sum_i (y_i - \bar{y})^2$.
- **Coefficient of determination:** $R^2 = 1 - \text{RSE}$.

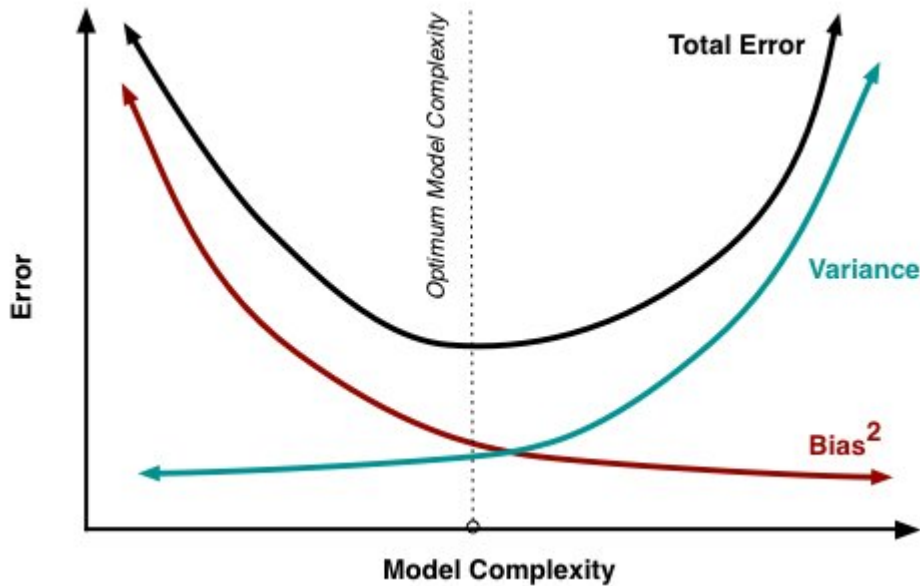
Tuning Model Complexity: Bias/Variance Dilemma

- Consider $y = f(x) + \varepsilon$, where $\varepsilon \sim N(0, \sigma^2)$ and \hat{f} is an estimator of f .
- **Mean squared error (MSE)** of a model: $E[(y - \hat{f})^2] = E[(f - \hat{f} + \varepsilon)^2] = E[(f - \hat{f})^2] + 2E[(f - \hat{f})\varepsilon] + E[\varepsilon^2] = E[(f - \hat{f})^2] + E[\varepsilon^2] = b_f(\hat{f})^2 + r_f(\hat{f}) + \sigma^2 = \text{bias}^2 + \text{variance} + \text{noise}$.
- **Bias/variance dilemma:** Models with a lower bias in parameter estimation have a higher variance of the parameter estimates across samples, and vice versa.
- In a sense, high bias implies **underfitting** and high variance implies **overfitting**.

Model Selection Procedures

- In practice, we cannot calculate the bias and variance for a model, but we can calculate the total error.
- **Cross-validation:**
 - The validation error is an estimate of the total error except that it also contains the variance of the noise.
 - Cross-validation makes no prior assumption about the model or parameters.
- **Regularization** introduce an **augmented error function** to penalizes complex models with large variance.
- The augmented error function can be seen as an **optimism** estimating the discrepancy between training and test error.
- The weight of the penalty λ is optimized using cross-validation.
- **Akaike's information criterion (AIC)** and **Bayesian information criterion (BIC)** work by estimating the optimism and adding it to the training error to estimate test error, without any need for validation.
- **Structural risk minimization (SRM)** uses a set of models ordered in terms of their complexities.
- **Minimum description length (MDL)** is based on an information theoretic measure.

- **Bayesian model selection** is used when we have some prior knowledge about the appropriate class of approximating functions.



Multivariate Methods

- Multivariate Data
- Parameter Estimation
- Estimation of Missing Values
- Multivariate Normal Distribution
- Multivariate Classification
- Tuning Complexity
- Discrete Features
- Multivariate Regression

Estimation of Missing Values

- **Imputation**: the process of replacing missing data with substituted values.
 - **Mean imputation** substitutes the mean (average) of the available data for that variable in the sample.
 - **Imputation by regression** predicts the value of a missing variable from other variables whose values are known for that case.

Multivariate Normal Distribution

- **Mahalanobis distance**: $(x - \mu)^\top \Sigma^{-1} (x - \mu)$.

- The projection of a d -dimensional normal on the vector w is univariate normal.
- Suppose $x \sim N(\mu, \Sigma)$. Then, $w^\top x \sim N(w^\top \mu, w^\top \Sigma w)$.

Multivariate Classification

- Assume that the feature space is D -dimensional.
- The discriminant function for class $i \in \{1, \dots, K\}$: $g_i(x) = \log P(x|y=i) + \log P(y=i)$.
- Assume $P(x|y=i) \sim N(\mu_i, \Sigma_i)$.
- $g_i(x) = -\frac{D}{2} \log 2\pi - \frac{1}{2} \log |\Sigma_i| - \frac{1}{2} (x - \mu_i)^\top \Sigma_i^{-1} (x - \mu_i) + \log P(y=i)$.
- $\mu_i \sim m_i$ and $\Sigma_i \sim S_i$ are estimated from N observations using maximum likelihood estimation.
- **Quadratic discriminant analysis (QDA):**
 - $g_i(x) \propto x^\top W_i x + w_i^\top x + b_i$, where
 - $W_i = -\frac{1}{2} S_i^{-1}$.
 - $w_i = S_i^{-1} m_i$.
 - $b_i = -\frac{1}{2} \log |S_i| - \frac{1}{2} m_i^\top + \log \hat{P}(y=i)$.
 - The decision boundary is a quadric hypersurface in D -dimensional space.
 - The number of parameters:
 - KD for the means.
 - $KD(D+1)/2$ for the covariance matrices.
- **Linear discriminant analysis (LDA):**
 - Assumption(s):
 - Covariance matrix for each class is shared, i.e., $\Sigma = \Sigma_i$ for class $i \in \{1, \dots, K\}$.
 - $g_i(x) \propto w_i^\top x + b_i$, where
 - $w_i = S^{-1} m_i$.
 - $b_i = -\frac{1}{2} m_i^\top S^{-1} m_i + \log \hat{P}(y=i)$.
 - The number of parameters:
 - KD for the means.
 - $D(D+1)/2$ for the shared covariance matrix.
- **Naïve Bayes' classifier:**
 - Assumption(s):
 - Covariance matrix for each class is shared, i.e., $\Sigma = \Sigma_i$ for class $i \in \{1, \dots, K\}$.
 - Independent variables, i.e., Σ is diagonal.
 - The number of parameters:
 - KD for the means.
 - D for the shared variances.
- **Euclidean distance classifier:**
 - Assumption(s):
 - Covariance matrix for each class is shared, i.e., $\Sigma = \Sigma_i$ for class $i \in \{1, \dots, K\}$.

- Independent variables, i.e., Σ is diagonal.
- Equal variances, i.e., $\Sigma = \sigma^2 I$.
- The number of parameters
 - KD for the means.
 - 1 for the shared variance.
- **Nearest centroid classifier:**
 - Assumption(s):
 - Covariance matrix for each class is shared, i.e., $\Sigma = \Sigma_i$ for class $i \in \{1, \dots, K\}$.
 - Independent variables, i.e., Σ is diagonal.
 - Equal variances, i.e., $\Sigma = \sigma^2 I$.
 - Equal priors, i.e., $P = P(y = i)$ for class $i \in \{1, \dots, K\}$.
 - The number of parameters:
 - KD for the means.
 - 1 for the shared variance.

Tuning Complexity

- **Regularized discriminant analysis (RDA):**
 - Substitute covariance matrix for class S'_i is the sum of three weighted components:
 - $\alpha s^2 I$: identity matrix.
 - βS : shared covariance matrix.
 - $(1 - \alpha - \beta) S_i$: class-specific covariance matrix.
 - Consider three scenarios:
 - $\alpha = \beta = 0$: quadratic discriminant analysis (QDA).
 - $\alpha = 0$ and $\beta = 1$: linear discriminant analysis (LDA).
 - $\alpha = 1$ and $\beta = 0$: nearest centroid classifier.
 - α and β are optimized by cross-validation.

Dimensionality Reduction

- [Introduction](#)
- [Subset Selection](#)
- [Principal Component Analysis](#)

Introduction

- The complexity depends on the number of input dimensions.

- Two main methods for reducing dimensionality:
 - **Feature selection:** finding k of the d dimensions.
 - **Feature extraction:** finding a new set of k dimensions that are combinations of the original d dimensions.
- Categories of *feature extraction* methods:
 - *Unsupervised:*
 - *Linear:* principal component analysis, factor analysis, multidimensional scaling
 - *Nonlinear:* isometric feature mapping, locally linear embedding, Laplacian eigenmaps
 - *Supervised:* linear discriminant analysis,

Subset Selection

- Two approaches: **forward selection** vs **backward selection**.
- Subset selection is *supervised*.

Principal Component Analysis

- Given data x_1, \dots, x_n , with $\text{Cov}(x) = \Sigma$.
- PCA is an optimization problem that maximizes the variance of the projection of x on the direction of w , where w is a unit vector.
- Mathematical definition: maximize $\text{Var}(w^\top x) = w^\top \Sigma w$ subject to $w^\top w = 1$.
- Lagrange problem: maximize $w^\top \Sigma w - \lambda(w^\top w - 1)$.
- Taking the derivative with respect to w and setting it equal to 0, we have $\Sigma w = \lambda w$.
- The principal component is the eigenvector of the covariance matrix with the largest eigenvalue.
- **Proportion of variance** explained by the k principal components is $\sum_{i=1}^k \lambda_i / \sum_{i=1}^d \lambda_i$.
- **Scree graph:** the plot of explained variance as a function of the number of eigenvectors kept.
- **Karhunen-Loève expansion** allows using class information.
- **Common principal components** assumes:
 - The principal components are the same for each class.
 - The variances of these components differ for different classes.