Rangkuman UTS Machine Learning

BAB I – Introduction

* **Machine Learning** adalah salah satu disiplin ilmu dari computer science yang mempelajari bagaimana membuat komputer/mesin itu mempunyai suatu kecerdasan
* **Machine learning** memiliki tiga maca learning process
  + **Supervised learning**
  + **Unsupervised learning**
  + **Reinforcement learning**
* Aplikasi machine learning dalam dunia nyata adalah
  + Interface (speech recognition, NLP)
  + AI (games komputer, entertainment, robotik)
  + Data analysis (information retrieval, data mining)
  + Bilogikal (Perangkaian gen, komputasi farmalogi)
  + Computer (optimisasi run time)
  + Industrial (Diagnosa error)
  + Dll
* Dalam machine learning ada beberapa topik yang dibahas, yaitu
  + **Concept Learning**
    - Merupakan metode learning dari contoh yang sudah ada
    - Hipotesa urut mulai general ke yang lebih spesifik
    - Jika data yang digunakan semua positif menggunakan **find-s**
    - Jika data yang digunakan adalah keduanya baik positif maupun negatif menggunakan **candidate elimination**
    - Data harus konsisten
    - Jawaban hanya ada dua kemungkinan, ya atau tidak
  + **Bayesian Learning**
    - Proses learning berasal dari contoh/examples
    - Termasuk ke dalam supervised learning
    - Didasarkan pada teorema bayes
    - Menggunakan data positif maupun negatif
    - Data tidak harus selalu konsisten
    - Jawabannya ditunjukan dalam bentuk probabilitas
    - Digunakan untuk fungsi-fungsi klasifikasi
  + **Clustering**
    - Tidak memiliki jawaban dalam data
    - Termasuk ke dalam unsupervised learning
    - Hanya membuat suatu klasifikasi tanpa label/jawaban
    - Labelisasi (penentu jawaban) adalah tanggung jawab user
  + **Neural Network**
    - Mensimulasikan kerja otak manusia
  + **Genetic Algorithm**
    - Algoritma yang memanfaatkan proses seleksi alamiah, yaitu proses evolusi
    - Dalam proses evolusi, individu secara terus-menerus mengalami perubahan gen untuk menyesuaikan dengan lingkungan hidupnya.
    - Proses seleksi ini melibatkan perubahan gen melalui proses perkembangbiakan
  + **Reinforcement Learning**
    - Proses learning berasal dari pengalaman
    - Menggunakan konsep reward dan punishment dalam proses learning
  + **Dll**

BAB II – Concept Learning

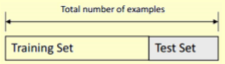
* **Find-S,** adalah suatu metode paling sederhana yang dapat digunakan untuk mendapatkan suatu hipotesa berdasarkan data, metodenya adalah dengan mencari kesamaan nilai atribut untuk memperoleh suatu hipotesa.
  + **Kelemahannya** adalah data yang digunakan harus bersifat konsisten dan tidak bias (data tidak masuk akal) & mengabaikan negative data
  + **Kelebihannya** adalah sangat sederhana
* **Candidate-Elimination**, suatu metode dimana data positif dan negatif dijadikan variabel dan dikalkulasi untuk menguatkan hipotesa
  + **Kelemahannya** adalah jika data tidak konsistem S dan G tidak akan sama, dan sangat sulit diimplementasikan pada programming
  + **Kelebihannya** adalah data negatif juga dipertimbangkan untuk menghitung hipotesis

BAB III – Algoritma Nearest Neighbor

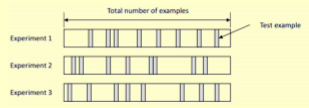
* **Nearest Neighbor** merupakan suatu metode untuk mengklasifikasikan suatu data baru berdasarkan similaritas dengan labeled data yang sudah ada
* **Satuan jarak** umumnya menggunakan euclidian
* Algoritma NN juga memiliki nama lain, yaitu
  + Lazy algorithm
  + Memory-based
  + Instance-based
  + Exemplar-based
  + Case-based
  + Experience-based
* Ada dua jenis algoritma NN, yaiut
  + 1-NN
    - Pengklasifikasian dilakukan terhadap 1 label data terdekat
    - Algoritma 1-NN
      * Hitung jarakan antara data baru ke setiap labeled data
      * Tentukan 1 labeled data yang mempunyai jarak yang paling minimal
      * Klasifikiaskan data baru tersebut ke dalam labeled data tersebut
  + k-NN
    - Pengklasifikasian dilakukan terhadap k label data terdekat
    - K > 1
    - Algoritma k-NN
      * Tentukan k
      * Hitung jarak antara data baru ke setiap labeled data
      * Tentukan k labeled data yang mempunyai jarak yang paling minimal
      * Klasifikasikan data baru ke dalam labeled data yang mayoritas (jika terdapat lebih dari satu label yang sama)
  + Keuntungan Algoritma Nearest Neighbor
    - Analytically tractable, dapat ditelusuri analisanya
    - Implementasinya sangat sederhana
    - Tingkat error lebih besar dibandingkan bayesian, tapi lebih kecil dibandingkan 2 x bayesian
  + Kelemahan
    - Membutuhkan memori yang besar
    - Harga komputasi juga besarr

BAB IV – Validation model

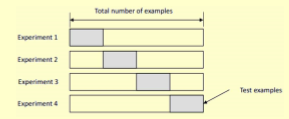
* Validation model digunakan untuk mencari tingkat error suatu metode klasifikasi (seperti NN), pada dataset yang spesifik
* Terdapat beberapa validation model, antara lain adalah,
  + Holdout method



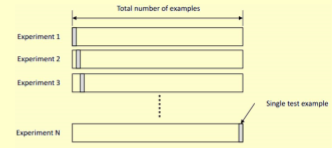
* + - Holdout membagi dataset menjadi dua grup,
      * Training set – digunakan sebagai training
      * Test set – digunakan untuk menentukan nilai error
    - Kelemahan
      * Jika dataset yang dimiliki acak-acakan, maka susah untuk mencapai hasil maksimal untuk testingnya
      * Nilai error akan dinilai salah jika pembagian datanya “tidak baik”
  + Random subsampling



* + - Dataset dibagi ke dalam *x* data sebanyak *n* (setiap segmen *x* berisi data berjumlah *n* data) dan diambil menjadi data testing **tanpa pengganti**, sisanya menjadi data trainign
    - Setiap data testing akan dilakukan test terhadap data training yang tersisa untuk mendapatkan error per eksperimen
    - Eksperimen dilakukan sebanyak *k* kali yang sudah ditentukan sebelumnya
    - True error didapatkan dari nilai rata rata error per eksperimen
    - Nilai errornya lebih akurat dibandingkan *holdout method*
  + K-fold cross validation



* + - Dataset dibagi sama rata sebanyak *k* kali. Pendefinisian data testing dan data training dilakukan per eksperimen secara bertahap pada setiap pembagiannya
    - Untuk setiap *k* eksperimen, digunakan *k-1* fold sebagai data trainign dan lainnya sebagai data testing
    - Kelebihannya adalah digunakannya seluruh dataset sebagai data testing dan data training
    - True error didapatkan dari nilai rata rata error per eksperimen
  + Leave-one-out cross validation



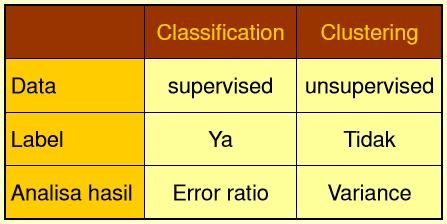
* + - Eksperimen dilakukan sebanyak jumlah dataset (*n)*
    - Nilai error dihitung per eksperimen dengan definisi data testing adalah data tunggal ke-*n* pada setiap iterasi eksperimen
    - True error didapatkan dari rata rata error per eksperimen
  + Bootstrap



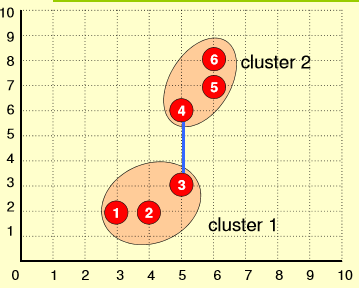
* + - Teknik bootstrap melakukan resampling disertai dengan replacement
    - Data testing dipilih secara random sebanyak *x-random* kali, setiap data yang diambil sebagai data testing akan di-replace dengan random data yang masih tersisa
    - Proses ini dilakukan sebanyak *k* kali eksperimen
* Berapa FOLD yang dibutuhkan unuk mencapai hasil paling optimum ?
  + Jika menggunakan jumlah fold tinggi
    - + Bias akan semakin kecil (semakin akurat)
    - - Komputasinya lama
    - - Variansi True error akan semakin besar
  + Jika menggunakan jumlah fold besar
    - +waktu komputasi semakin cepat
    - + Variansi True erorr semakin kecil
    - - Bias estimatornya besar
  + Pada prakteknya
    - Jika diterapkan pada aplikasi yang mebutuhkan akurasi tinggi, disarankan untuk menggunakan jumlah fold yang tinggi, karena hasilnya akurat meskipun membutuhkan resource komputasi yang besar (contoh aplikasi kedokteran)
    - Jika diterapkan pada aplikasi yang membutuhkan waktu komputasi rendah, gunakan fold yang kecil (aplikasi pencarian)
  + Pilihan yang umum, nilai foldnya adalah K=10
* Variansi - Selisih antara masin-masing error pada error testing
* Bias – jarak antara True error dan rata-rata error

BAB V – Clustering

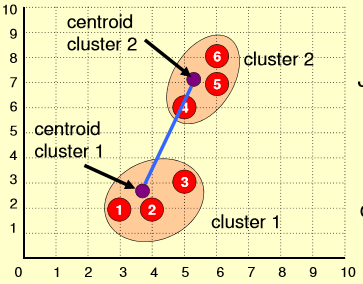
* **Cluster** adalah kumpulan objek yang “sama” dengan sesamanya, dan “tidak sama” dengan object yang berbeda cluster
* **Clustering** adalah proses untuk mengorganisir object menjadi beberapa grup yam anggotanya memiliki karakteristik yang sama
* **Clustering vs Classification**



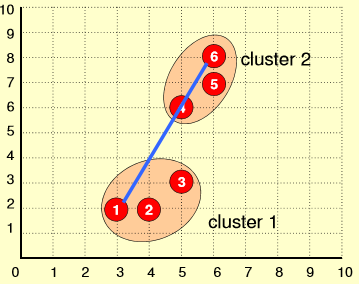
* Clustering dapat dilakukan menggunakan beberapa metode, yaitu
  + Decision Tree
  + K-means
  + Single Linkage
  + Complete Linkage
  + Average Linkage
  + Dll
* Clustering memiliki beberapa karakteristik, yaitu
  + Partitioning clustering
    - Disebut juga exclusive clustering
    - Setiap data harus termasuk ke cluster tertentu
    - Memungkinkan untuk setiap data berpindah cluster pada saat tahapan proses yang baru
    - Contoh : K-means, residual analysis
  + Hierarchical clustering
    - Setiap data harus masuk ke cluster tertentu
    - Data tersebut tidak dapat berpindah ke cluster lainnya pada saat tahapan proses berikutnya
    - Contoh : Complete Linkage, Average Centroid
  + Overlapping clustering
    - Setiap data dapat termasuk ke lebih dari satu cluster
    - Data mempunyai **nilai keanggotaan** untuk beberapa cluster
    - Contoh : Fuzzy C-means, Gaussian Mixture
  + Hybrid
    - Gabungan antara karakteristik partitioning, hierarchical dan overlapping
* **K-means** termasuk partitioning clustering yang memisahkan data ke *K* daerah/cluster yang terpisah
  + **K-means** sangat terkenal karena kemudahan dan kemampuannya untuk melakukan clustering data besar dan data outlier dengan sangat cepat
  + Algoritma K-means
    - Tentukan *k* sebagai jumlah cluster yang ingin dibentuk
    - Bangkitkan *k* centroids (titik pusat cluster) awal secara random
    - Hitung jarak setiap data ke masing-masing centroid
    - Tentukan posisi centroid baru dengan cara menghitung nilai rata-rata dari data-data yang memilih pada centroid yang sama
    - Kembali ke langkah tiga, jika posisi centroid baru dengan centroid yang lama tidak sama atau lebih besar dari batas error yang didefinisikan
  + Karakteristik
    - Sangat cepat dalam proses clustering
    - Pembangkitan centroid awal secara random memegang peranan penting jalannya proses K-means
    - Memungkinkan adanya cluster yang tidak memiliki anggota
    - Hasil clustering K-means selalu berubah-ubah, terkadang baik, terkadang buruk
    - K-means sangat sulit untuk mencapai global optimum
* Hierarchical clustering memiliki beberapa tipe, yaitu divisive dan agglomerative
  + Divisive
    - Dimulai dari satu cluster besar menjadi *k* cluster
    - Pembagian dari atas ke bawah
  + Agglomerative
    - Dimulai dari *n* cluster menjadi *k* cluster
    - Modelnya adalah penggabungan dari cluster masing-masing data ke cluster yang lebih sedikit
  + Algoritma Hierarchical Clustering
    - Tentukan *k* sebagai jumlah cluster yang ingin dibentuk
    - Setiap data dianggap sebagai cluster, jika N = jumlah data, dan *n*=jumlah cluster, maka *n=*N
    - Hitung jarak antar cluster
    - Cari 2 cluster yang mempunyai jarak antar cluster yang paling kecil dan gabungkan kedua data/cluster tersebut menjadi satu cluster
    - Jika *n*>k, maka kembali ke langkah 3
  + Hierarchical clustering memiliki 4 metode penentuan similiarity,
    - Single Linkage



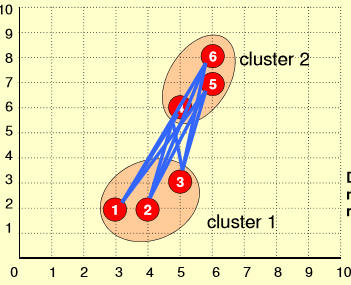
* + - * Jarak minimum antara dua cluster (atau jarak minimum antara dua anggota yang berbeda cluster
      * Cocok digunakan untuk kasus clustering yang shape independent, karena kemampuannya untuk membentuk pattern tertentu dari cluster
    - Centroid Linkage



* + - * Jarak minimum dua cluster berdasarkan centroidnya
      * Baik digunakan untuk kasus clustering dengan normal data set distribution, namun tidak cocok jika datanya banyak mengandung outlier
    - Complete Linkage



* + - * Jarak paling jauh diantara dua anggota cluster yang berbeda
      * Baik untuk memperkecil variansi di dalam cluster. Metode ini juga baik untuk data yang memiliki outlier
    - Average Linkage



* + - * Jarak rata rata dari masing-masing anggota dua cluster yang berbeda
      * Metode terbaik untuk seluruh kasus, namun waktu komputasinya paling tinggi jika dibandingkan dengan metode hierarchical yang lain.