1. Week - 30 September 2024 Monday

Derslerimizi Prof.Dr. Mine Elif KARSLIGİL hocamız ile işliyoruz.

elif@yildiz.edu.tr
haftanın 5 günü sabahtan akşama üniversitede oluyor

* %10 1. ödev
* %15 2. ödev
* %15 3. ödev
* %20 vize
* %20 dönem projesi (Final Project)
* %20 final

- Ödevlerde kendi başımıza kodlayabilecek durumda olmamız isteniyor.
- Ödevlerin python üzerinde yapılması istenecek.

Dönem projesi (Final Project) * Proposal: 1 page - Oct 21 * Choose a topic that you are interested in * Progress Report: 2-3 pages: November 18 * Presentation: Last week of the semesters * At least 6 pages in journal paper format - Last week of the semester * Oturup da Support Vector Machine'yi kodlamamız istenmeyecek. Fakat ufak tefek kodlamaları kendimiz yapabiliriz. * Bir data bulamıyorsanız kaggle üzerinden yarışma datasetleri üzerine çalışabilirsiniz. * Örnek: Trendyol mesajları üzerinden bir proje yapılabilir. * Feature extraction * Kargo yorumu olanları ayırdım * Memnuniyet yorumu olanları ayırdım * Memnuniyetsizlik yorumu olanları ayırdım * Gereksizleri temizledim gibi gibi * TODO: Sermaye arttırımı veya kap a bildirilen yorumlar üzerinden bir proje yapılabilir. * TODO: Loglar üzerinden makine öğrenmesi ile anomally detection yapılabilir. * Introduction, Related Work, Methodology, Results, Conclusion gibi sınıflandırarak makale formatında yazılmalı.

Course Information

Derin öğrenme 2011 yılından sonra çok popüler oldu. Klasik makine öğrenmesi teknikleri biraz geri planda kalmaya başladı. * Derin öğrenmede özellik sayısını biz belirlemiyoruz. Elindeki datadan kendisi çıkartıyor. * Karakteristik özellikleri çıkarabilmesi için modele yardımcı olmak gerekebilir. (Feature Engineering)

Bu ders klasik makine öğrenmesi dersidir.
Derin öğrenme dersi bölümde farklı bir ders olarak verilmektedir.
Ödevler online.yildiz.edu.tr üzerinden yüklenecek.

Kitaplar * Introduction to Machine Learning 2nd Edition by Ethem Alpaydın, 2010 * Pattern Recognition and Machine Learning by Christopher Bishop, 2006 * Deep Learning by Ian Goodfellow, Yoshua Bengio, Aaron Courville, 2016 * Machine Learning: A Probabilistic Perspective by Kevin P. Murphy, 2012 * Machine Learning Yearning by Andrew Ng, 2018

Prerequisites * Linear Algebra * Probability Theory * Design and Analysis of Algorithms * Coding in Python

What is Artificial Intelligence?

Some Subgoals of AI * Perception: Vision, Speech, NLP, smell, touch, taste * Reasoning: Planning, Problem Solving, Decision Making: Getting logical conclusions and making prodiction * Search: The process of navigating from a starting state to a goal state by transitioning through a series of intermediate states * Communication: Providing human-computer interaction. * Learning: The ability to improve performance based on experience (**by Machine Learning**)

İnsan gibi davranan bir sistem yapıyorsan, öğrenme özelliği olmalı.

TODO: İncele. Stanfort University, Fei Fei Li * Spor ayakkabı tanıtma: bağcıklı, bağcıksız, bot etc. Bir sürü değişken var. * ImageNet diye bir şey yapıyor. * Milyon tane görüntüden bu nedir diye bir soru ile insanlara etiketletiyor. * Her sene bir yarışma düzenliyorlar. * %70 - %71 başarı çıkıyor * Kanada da bir ekip blackberry ye ses işleme satamadığı için, kodu ImageNet yarışmasına uyarlıyorlar ve bir anda 15 puan gibi artarak %85 başarıya çıkıyor. Bu yöntem AlexNet adını alıyor.

Traditional Programming vs Machine Learning

- Traditional: Input + Program = Output
  - Web Page Coding
  - Bank ATM Software
  - Tetris Game
  - Stock Control Software
  - Deterministic olarak çalışır. Her zaman aynı input için aynı output u verir.
- Machine Learning: Input + Output = Program
  - Is there a cat in the image?
    - Kedinin gözü bağlı olabilir.
    - Ters yatmış olabilir
    - Yan duruyor olabilir.
    - Farklı ırkta olabilir.
    - Farklı renkte olabilir.
    - Bu durumların hepsi için ayrı ayrı if else program yazmak yerine, Machine Learning ile bir model oluşturulur ve bu model ile tahmin yapılır.
    - Yazın sadece yazlığa gittiğinde yemek siparişi veriyor olabilir.

Kullanım Alanları * Sound Asistant * Siri, Google Assistant, Alexa, Cortana * Recommendation Systems * Netflix, Amazon, Spotify * Size benzeyen kişilerin beğendiği filmleri önerir. * Aşırı öğrenme nedeniyle sizin tarzınızı bilir. Yeni başlayan şeyler önerilmemeye başlayacak. Buna Cold Start Problem deniyor. * Customer Segmentation (Unsupervised Learning) * Müşterileri segmentlere ayırır. * Örneğin, 20-30 yaş arası, 30-40 yaş arası, 40-50 yaş arası gibi. * Veya kullandıkları ürünlere göre segmentlere ayırabilir. * Automatic Chatbot * Müşteri hizmetleri * Sipariş takibi * Randevu alma * Bilgi alma * Chat GPT * Çok güçlü donanımlar üzerinde çalışıyor. * Çok büyük data üzerinde çalışıyor. * Bizim kendi çabalarımızla yapacağımız bir geliştirme bununla yarışamaz duruma geldi.

Types of Machine Learning

Supervised Learning (Predictive - Predict the Feature)

Build a predictive model based on labeled (example) data. Etiketlenmiş veriler (training set) üzerinden yeni gelen verilerin hangi sınıfa ait olduğunu tahmin eder. * Classification * Binary Classification * Multi-Class Classification * etc. * Regression * Linear Regression * Continuous output * Discrete output * etc. ## Unsupervised Learning

Benzerliklerine göre verileri gruplandırır. Verinin sınıfı belli değil. * Clustering * Association * Dimensionality Reduction * Reinforcement Learning * etc.

Example: Credit Card Fraud Detection

- Veri kümesi toplayabiliyor muyum?
- Toplayabiliyorsam ne kadar veriyi ne kadar sürede toplayabilirim?
- Bu konunun gerçekten yapay zeka ile çözülmesi gerekiyor mu gerekmiyor mu?
  - Gerçekten hiç kullanılamayacaksa gerek yok.

Ön İşleme * Datanın anonim hale getirilmesi * Normalizasyon * Sayısal olmayan datayı sayısal hale getirme

- Kaç kişi etiketleyecek
  - 2 farklı doktor bir görüntü üzerinde farklı yorumları olabilir.
  - Göreceli cevabı olan bir durumda 100 kişiye etiketletip bir threshold (%70 gibi) belirleyebiliriz.
    - Eğer %70'in üzerinde oy almışsa o sınıfa ait olduğunu kabul edebiliriz.
    - Eğer %30'un altında oy almışsa o sınıfa ait olmadığını kabul edebiliriz.
    - %30 ve %70 arasında oy alanları eğitim için kullanma ki net sonuçlar elde edilsin.

- Determining Learning Approach:
  - Can the data be labeled?
    - Yes: Supervised Learning (FRAUD, NORMAL)
    - No: Unsupervised Learning
- Data Set Preparation
- Assumptions
- Feature Selection - Extraction
- Model Selection
- Model Training (Evaluation)

Datayı 2'ye ayırma. Her zaman 3'e ayırmak daha iyidir.

Data Set: Training Set, Validation Set, Test Set * Training Set: Modelin eğitildiği veri seti * Validation Set: Modelin eğitilirken doğrulama yapılması için kullanılan veri seti * Test Set: Modelin eğitildikten sonra test edilmesi için kullanılan veri seti. Hiç görmediği veri üzerinde test edilir.

Makale yazarken * Makine öğrenmesi ve deep learning kullanılmıştır biraz yanlış ifade oluyor. * Makine öğrenmesinin bir alt dalı olan derin öğrenme kullanılmıştır gibi söyleyebiliriz.

Aşağıdaki kümeler birbirleri ile ilişkilidir. * Artificial Intelligence: * Machine Learning: * Deep Learning: * Data Mining: * Data Science: Ben bu veriyi yorumlayayım. Burdan ne çıkarabilirim? * Big Data:

2. Week - 7 October 2024 Monday

Bu hafta karar ağaçları ile ilgili konuları işleyeceğiz.

Strategies of a machine learning model:

Real World -> Measuring Device -> Preprocessing -> Feature Selection -> Model Selection -> Model Training -> Model Evaluation -> Model Deployment

TODO: Bu dersin sonunda entropi (entropy) ve bilgi kazancı (information gain) hesaplamayı öğren.

Data toplarken genel bir sistem analizi yapıp, gözle gördüğünüzün dışında da data toplamanız gerekmektedir. * Erişebildiğiniz ve lazım olup olmadığından emin olmadığınız her şeyi toplayabilirsiniz. * Bu bilgileri feature extraction ile zaten gerekirse drop edeceğiz. * Principal Component Analysis: En çok kullanılan özellik azaltma yöntemlerinden biridir. * Eigen Value ve Eigen Vector yani matrisin önemli olan özellikleri üzerinden yüz tanımayı yapalım denmiş. Sonrasında yüz tanımanın başarımı artmış. * Eskiden göz, burun, ağız gibi yüzün ayırt edici kısımlarının koordinatlarının bir birine mesafesine bakılarak yüz tanıma yapılmaya çalışılıyordu. Bu çok başarılı bir yöntem değildi.

Unsupervised Learning

- Çoğu zaman bizde sınıfı bilmiyoruz.
- Bazende milyonlarcad veya milyarlarca veri var ve bunları manuel etiketlememiz mümkün değil.
- Temelde yapılan iş aslında veriyi özelliklerine göre gruplandırmak.
- Kilo ve yaş grubu verilen bir veri setinde kız veya erkek çocuk olup olmadığını tahmin etmek.

D:
x: input features

Supervised Learning

Inputlarımız datalar, Elif'in ve Ayşe'nin yüzü şeklinde etiketli.
Outputlarımız ise etiketler, Bu yeni fotoğraftaki kişi Elif veya Ayşe dir bilgisi.

D:<x, y>
x: input features
y: output label

- Bugün supervised learning in içindeki karar ağaçlarını (decision trees) işleyeceğiz.
- Örnek sayısının çok fazla olmadığı zamanlarda karar ağaçları çok kullanışlı ve iyi sonuçlar veren bir yöntemdir.

İki farklı yöntem

Regression: Sürekli zamanda bir bilgi ediniliyor. İleriye yönelik bir tahmin yapmaya çalışıyoruz gibi düşünebiliriz. * Geçtiğimizi 10 yılda alınan yağış şu kadar iken fındık 100 ton du. 2024 yılında xyz yağış yağdı, kaç ton fındık alınabilir? * Ocak ayından bu yana beşiktaş ta ev fiyatları şu kadardı. Aralık ayına geldiğimde fiyat ne kadar olacak?

Classification: * kedidir, köpektir gibi sınıflandırıyoruz.

Two Approaches;
Discriminative Classifiers: * Decision Boundry (Karar sınırı): Sınıflandırma yaparken sınıfların birbirinden ayrıldığı çizginin yerini belirlememiz lazım. * Spam not spam sınıflandırması yaparken spam ve not spam arasında bir çizgi çizmemiz gerekiyor.

Generative Classifiers: * Bayes Classifier * Verilen bilgilerden örneğin hangi sınıfa ait olabileceği bilgisini çıkartıyorlar. %80 A sınıfı %20 B sınıfı gibi.

En temellerinden bir tanesi Binary Classification. Sadece 2 sınıfa ayırma yapılır. Spam ve not spam gibi.
Multi-Class Classification: Birden fazla sınıf varsa. Köpeğin türü, labrador, golden retriever, rado gibi.

Decision Stamp: Tek bir özellikle sonuca varabiliyorsanız budur.

Desicion Trees

2 önemli avantajı vardır. * En önemli özelliği embedded feature selection yapıyor olmasıdır. * Ağacı oluştururken gereksiz özellikleri kendisi eliyor. * Ayrı bir özellik seçme algoritması kullanmamıza gerek yok. * Information Gain bu işe yarıyor. En önemli özellikler ön plana çıkıyor. * Ağacı oluşturduktan sonra altta bulunan node ları prune (keserek) ederek bilgi kazancı tekrar hesaplanır * Ağacın daha iyi mi sonuç verdiğine bakılır. * İyi sonuç verdiyse bu şekliyle desicion tree oluşturulmuş olur. * Karar sınırları çok esnektir. Yamuk yumuk olabilir. Sınıf başarısını pozitif etkilemektedir.

- Öğrenme karar ağacının oluşturulmasıdır
- Karar ağacını oluşturduktan sonra yeni gelen dataya bakıyorum ve karar ağacında ilerleyerek karar veriyoruz.
  - Hangi sınıfa ait olduğuna karar verilir (Classification)
  - Geleceğe yönelik tahmin olabilir (Regression)

Sayısal değerler olduğunda ne yapacağız? * Bir eşik seviyesi kullanmalıyız * Yaşı 40'dan büyükse şunu yap, küçükse bunu yap gibi. * Eşik seviyesini (threshold) belirlemek çok önemli. * Karar ağaçlarının en güzel özelliklerinden biri sparse data ile çalışabilmesidir.

Desicion tree yi programatic olarak kodlamak istersek aşağıdakine benzer bir durum ortaya çıkar; * Aslında klasik programlama gibi olmuş

if outlook = sunny AND humidity = normal
OR
....
then
play tennis

Ağacı Nasıl Oluşturacağımıza Nasıl Karar Vereceğiz?

- Köke ne koyduğumuz ağacın uzunluğunu (derinliğini) çok fazla etkiler.
- Non deterministic polinomial bir denklemdir. En iyi ağacı oluşturmak çok mümkün değil.
- Bu nedenle pruning yaparak ve farklı özellikleri kök düğüme koyarak denemeler yapmamız gerekiyor.
- Bölen özelliğimizin olabildiğince saf bir sınıf olmasına dikkat etmeliyiz.
- Saf olan özelliği bulmak için entropy hesabı yapılması gerekmektedir. Entropy sıfıra yaklaştıkça saflık artar. 0 iken özellik tamamen saftır.

Karar ağacını oluştururken entropileri de kullanarak bilgi kazancı hesabı yapıyoruz. Bilgi kazancı en yüksek olan özelliği kök olarak seçiyoruz. * Alt node lardaki hesaplamalarda benzer şekilde yapılıyor.

Entropy

Karar ağacına hangi özelliği root düğüm olarak yerleştireceğimize karar vermede kullanılır.

Information Gain (Bilgi Kazancı)

Her alt kümenin entropisi hesaplanır.
Tüm veri setinin etropisinden çıkarılarak bilgi kazancı hesaplanır.

Overfitting in Decision Trees

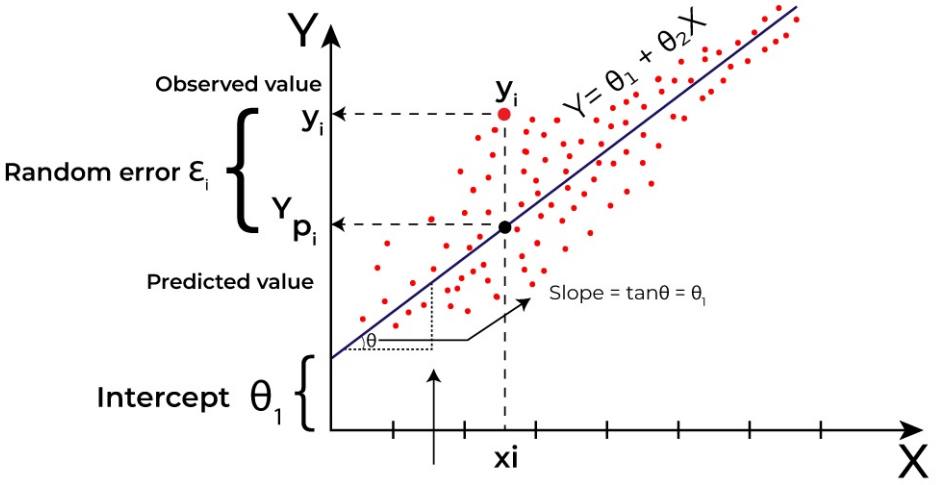
- Training de %90 başarı, testte %85 başarı varsa bu durum normaldir.
- Training de %90 başarı, testte ise %60 başarı varsa bu durum overfitting olabilir.
- Training set çok küçükse overfitting oluşabilir
- Pruning: Ağacı kesmek ve alt ağaçları ortadan kaldırmak anlamına geliyor. Prune ederken amacımız test ve validasyon datasında daha doğru sonuç elde etmektir.
- Çözüm
  - Reduced error pruning
  - Early stopping
  - Rule post pruning

3. Week - 14 October 2024 Monday

Lineer Regression konusu ile başlıyoruz.

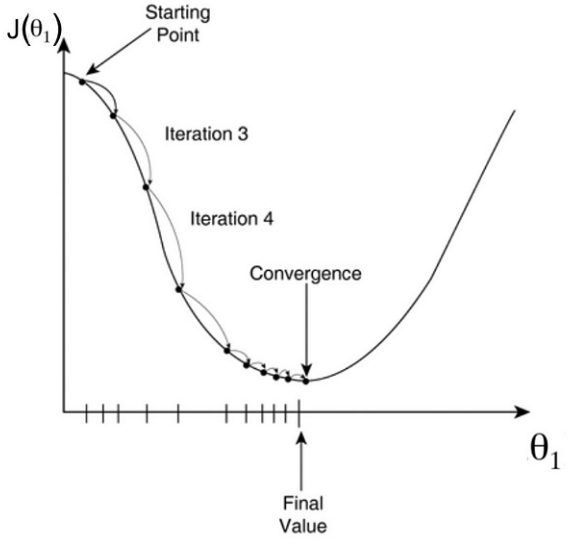
Linear Regression

Lineer bir doğru uydurarak bilinmeyenleri tahmin etmeye çalışmaktır. * Doğru örneklerin tam ortasından geçerek tüm veri setini temsil edebilecek şekilde olmalıdır. Bu tam olarak mümkün değil aslında.



Lineer Regression Error

- Öğrenme hatayı minimize edecek şekilde ağırlıkların değiştirilmesi demektir.
- Ağırlığı çok küçük seçersem iterasyon sayısı çok artarak global minimum a ulaşabiliyorum.
- Ağırlığı çok büyük seçersem iterasyon sayısı azalır fakat, global minimum a yaklaşamadan üzerinden pass geçebilirim.



Cost Function – "One Half Mean Squared Error":

$$J(\theta_0, \theta_1) = \frac{1}{2m}\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)^2$$

Objective:

$$\min_{\theta_0, \theta_1} J(\theta_0, \theta_1)$$

Derivatives:

$$\frac{\partial}{\partial\theta_0} J(\theta_0, \theta_1) = \frac{1}{m}\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)$$

$$\frac{\partial}{\partial\theta_1} J(\theta_0, \theta_1) = \frac{1}{m}\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right) \cdot x^{(i)}$$

Gradient Descent Algorithm (Iterative Local Optimization Algorithm)

- Gradient: Derivative of a function that has more than one input feature
- Aim: Find optimal weights that minimize the cost function

- Başlangıçta büyük adımlarla başlar ve sona yaklaştıkça eğim azalacağı için küçük adımlarla ilerler.
- Learning Rate başlangıçta büyük alınarak, zaman arttıkça küçültülebilir.

GD Algorithm

1 EPOCH: Tüm veri setini bir kere görmek anlamına gelir.

- Begin training with randomized weights and biases near zero +-0.5 REPEAT
  - STEP 1: Calculate the loss
  - Determine the **direction to move** the weights to **reduce the loss**
  - More the weights in that direction
  - Return to Step 1 UNTIL CONVERGENCE

convergence: minimum cost function value. convergence olana kadar bizim iterasyonlarımız devam edilebilir veya belirli bir iterasyon sayısına ulaşılınca durdurulabilir.

3 tane gradient descent türü işleyeceğiz. * Stochastic Gradient Descent * Batch Gradient Descent

Stochoistic Gradient Descent (Online Linear Regression)

- Online linear regression da deniyor.
- Her bir örnek için çıkışı hesapla
- Hesapladığın çıkış ile beklediğin çıkış arasındaki farkı hesapla
- Bu farkı minimize etmek için ağırlıkları güncelle

SGD Algorithm

- Her örnek için ağırlıkları ayrı ayrı hesaplayıp
- Ağırlıkları güncelliyoruz.
- Initialize weights
- Set learning rate
  REPEAT
  - STEP 1: Select a sample <xi, yi> from D
  - Calculate the loss
  - Update weights vector for feature Xj
  - wj(t + 1) = wj(t) - η * (yi - yi^) * xi
  - Return to Step 1 UNTIL CONVERGENCE

Batch Gradient Descent

- Tüm veri setini bir kere görüp hatayı hesaplıyoruz. (1 Epoch ilerlemek anlamına gelir)
- Sonra ağırlıkları güncelliyoruz.
- Tekrar tüm veri setini görüp hatayı hesaplıyoruz.
- Ağırlıkları güncelliyoruz.

Mini Batch Gradient Descent

- Tüm örnekleri görmek yerine, örneklerin alt kümesini alıp işlem yaparız.
- 1000 örenek varsa, mini batch size 100 örnek seçildiyse, 10 tane mini batch olur.
- 1 Epoch için 10 iterasyon gitmemiz gerekir.
- Bunda 100 örnek seçip, hatayı hesaplıyoruz.
- Sonra ağırlıkları güncelliyoruz.
- Tekrar 100 örnek seçip, hatayı hesaplıyoruz.
- Ağırlıkları güncelliyoruz.
- Bu şekilde devam eder.

Normalizations

Min-Max Scaling

Bunun sorunu aykırı değerlerin (outlier) öncesinde discard edilmesi gerekmektedir. * Yoksa outlier nedeniyle örneklerin scale i çok düşürülmüş olur. * Mesela sıcaklık 90 Derece olarak ölçüm yapılmış 1 tane, normal aralık 0 - 30 derece arasıysa bunu normalize ederken önemli verilerin distance farkını ignore etmiş oluyoruz.

Standard Deviation Normalization (Z-Score Normalization)

- Values are centered around mean. Böylelikle aykırı (outlier) değerlerin etkisi daha da azaltılmış olur.
- Xi normalize edilmemişse Xin büyük değerlerinden ötürü ağırlık hoplaya zıplaya gidiyor.

4. Week - 21 October 2024 Monday

Splitting Dataset

- Modelin başarısını ölçtükten sonra, (%83 çıktı) neden hata oluştu analizini yapmamız gerekiyor.

Veri setini 3 parçaya bölmeliyiz. * Training data * Validation (Development) data * Test data

Training Data

- Eğitim sırasında kullanılan datadır
- Başarım sonucunun en yüksek çıktığı yer burasıdır.

Validation (Development) Data

- Sonucunu bildiğimiz duruma göre başarı oluşturmaktır.
- Sonucu bilerek işlemleri yapıyoruz.
- Ağacı budamak gerekiyorsa buduyoruz.
- Parametreleri değiştirmek gerekiyorsa değiştiriyoruz.

- Yani aslında ince ayar (fine tuning) işlemlerini burada yapıyoruz.

Test Data

- Test seti başarısı o sistemin gerçek başarıdır.
- Test setin başarısı validation a yakın olursa modelimiz hazır diyebiliriz.
- Validation kısmında %80 başarı varken test setinde %60 gibi bir başarı elde ediliyorsa;
  - En başa geri dön. Training dataset ile tekrar eğit.
  - Validation ile validasyonunu yap.
  - Tekrar test datası ile sonuçları izle
  - Validation ve test birbirine yakın çıkana kadar devam et.

Başarı Ölçümü

Başarı ölçümünde accuracy ve response time gibi parametreler kullanılır.

Classifier	Accuracy	Running Time (ms)
A	0.90	80
B	0.92	95
C	0.95	1500

- Zaman kritik bir durumum yoksa en doğru sonucu ürettiği için C modelini seçerim.
- Eğer zaman kritik ise B'yi onun zamanı da çok geliyorsa A'yı seçerim.
- Sonuç olarak Accuracy kadar response time ında önemli olduğu uygulamalar olabilir. Çalışmalarda response time ıda göz önünde bulundurmak gerekebilir.
- Bir projeye başladığınızda, buna benzer projelerde performans ölçme metriği nedir diye bir bakın ve ona göre çalışmalarınızın performansını ölçüp karşılaştırın.
- Accuracy, precision, recall, hangisi kullanılacak?

Single Number Evaluation Metric:

Debugging the Learning Model

- Gather a sample of 100 dev set examples that the system misclassified and count what fraction of them are which class.
- Mutlaka ana senaryoları belirle ve çözümlerinin ne olacağı üzerine çalış.
- Genel bir sistemde konuya özel problemlerin sadece sık rastlananlarını belirle ve bunları çöz.
- Az rastlanan problemi çözmeye çalışmak overfitting e neden olabilir.
- Hataları katagorize etmeliyiz. Yöntemde bir şeyi değiştirirsem çözebilir miyim?
  - Kameranın yeri?
  - Tek kamera yerine 2 tane kamera?
- Cleaning up mislabeled examples
  - Sınıfları tanımlamayan fotoğrafları dataset ten çıkarmamız lazım. Örneğin yakınında bir yerde Lourve müzesi olarak etiketlenmiş örnek aslında müzeyi temsil etmemektedir. Bunu modelin doğru sınıflandırması mümkün değildir.
  - Etiketleme hatası çok sık rastlanan bir hatadır. Kime göre, neye göre bu şekilde etiketlenmiş?
  - Paper yazarken hakemlerde datanın kim tarafından etiketlendiğini sorar?
  - Genel bir dataysa kadın, erkek, yaşlar karışık şekillerde etiketletmek mantıklıdır.
  - Accuracy on dev set: 0.9
  - Errors due to mislabeled examples: 0.06
  - Errors due to other problems: 0.94

Large Validation (Dev) Set

Validation sette modelde %20 hata varsa aşağıdakini uygula; * Büyük bir kısmını otomatik etiketlet. * Küçük bir kısmını manuel olarak sen etiketle

- If you have a large dev set, split into two subsets

5000 dev set samples * 500 Eye ball -> Manually look at this * 4500 -> blackbox

Basic Error Analysis

- Build and train a basic system as quickly as possible.
- 100 örnek alarak küçük bir sistem kurup başarıyı ölçümle.
  - Buradan direkt hangi case lerde hata var görülebilir.
  - Bu veriye dayalı bir sonraki adımda nasıl ilerlemen gerektiğine karar ver.

Bias and Variance Trade Off

Bias: The error rate on the training set
Variance: How much worse the algorithm does on the dev (or test) set than the training set.

- Dev/Test set indeki performans training kümesindeki başarıdan küçüktür.
- Training set ile istediğiniz başarıya ulaşamıyorsanız training veri seti üzerinde algoritmanızı optimize etmeye çalışın.
- Training Er: 1%, Dev Error: %11 ise;
  - Bias: %1
  - Variance: %(11 - 1): %10: high variance
  - The classifier has very low training error, but it is failing to generalize to the dev set.
  - This is called **overfitting**.
  - Bias çok küçük ama variance çok büyük ise overfitting ile karşı karşıyayız demektir.
- If available bias is high, increase training dataset

Proje

kaggle+dataset+for+classic+machine+learning yazarak verisetini bulabilirsiniz.

- Yarışmalarda yapılanları tekrarlamayı deneyin
- Şu basit şeyleri yapın fakat bunlarla yetinmeyin tabi
  - batch size ını 500 den 1500 e kadar değiştirdimm.
  - parametreyi 0.1 den 1 e kadar değiştirdim.
  - learning rate i 0.1 den 1 e kadar değiştirdim.
  - Veri setiniz büyükse hepsini kullanmadan temel hataları çözün. Sonra tüm veriseti üzeirinde çalışın ki süreç hızlansın.

100 özellik vardı. Bunlardan * İnternetten 10 farklı yazarın 30 günlük yazılarını topladım. Yazarı tanıdım. gibi bir şeyde yapabilirsiniz. * İlk seferde seçtiğimiz konuyu beğenmezse bir hak daha vereceğim. * Fishing detection için bir veriseti oluşturulabilir belki.

5. Week - 28 October 2024 Monday

Sınavda formül çok sormuyor gibi. * Ezberlememizi istemiyor * Anlamamızı istiyor * Örnek verip yorumlamamızı istiyor.

Overfitting & Under Fitting

Underfitting occurs when a model is too simple. * Daha çok modelinizle alakalı problemler var demektir. * Veriyi temsil etmekte çok iyi değil. * İterasyonu erken kesmiş olabilirsiniz (epoch).

Model Selection

Her veri ve durum için en iyi sonucu veren bir model yoktur. Veriye ve context e göre hangi makine öğrenmesi modelini (desicion tree, linear regression, SVP, KNN, Naive Bayes, K-Means, ...) seçeceğimize biz karar vermeliyiz. * **Literatür Taraması**: Buna benzer sorunlar/konular için hangi modeller seçilmiş? Ne kadar başarı elde edilmiş? * **Model Selection**: İki tane aynı sonucu veren modeliniz varsa, basit olanı seçin. * **Feature Selection**: Modeli tasarlarken seçtiğiniz özellikler başarıyı çok etkiler. Hangi özellikleri dışarda bırakacaksınız? * **Parametre Seçimi**: Learning Rate, k değeri gibi çeşitli parametrelerin değerlerinin değişmesi modelin başarısını çok etkiler.

Fine Tuning & Measurement Steps

Aynı anda birden fazla şeyi değiştirmeyin. Böyle yaparsanız hangisinden kaynaklı başarının değiştiğini gözlemleyemezsiniz. * Özellikleri değiştirin, parametreleriniz aynı kalsın. Başarı ölçün. * Parametreleri değiştirin, özellikleriniz aynı kalsın. * Veri setini değiştirdiyseniz, başka bir şeyi değiştirmeden denemelerinizi yapın.

Train Test Splitting

1) Holdout Method

Şu anda çok sık kullanılan bir yöntemdir. Örnek sayısı çoksa direkt bu kullanılabilir.

- %70 training için, %30 testing için ayrılır.
- **Stratified Sample**: Advanced version of balancing the data
  - Make sure that each class is represented with approximately aqual proportions in both subsets
  - TODO: Fraud detection gibi nadir karşılaşılan veri için kullanacağımız veri setinde train ve test için bölüyorsam fraud olanların %70'i training %30'u test setinde bulunmalıdır. Random seçilmeli fakat oranları korunmalıdır.

Nadir veriler için; * **Data Augmentation**: Gerçekten çok gerçeğe benzer fakat gerçek olmayan datalar üretilerek nadir görülen sınıfın sample sayısı arttırılabilir. * **Percentage Changing**: Fraud olmayan datalar %90 ve fraud %10 ise * Yavaş yavaş data oranlarını değiştirin. * %80 normal ve %20 fraud denenebilir. * %60 normal ve %40 fraud gibi. * Bunu yaparken normal olanların sayısını azaltıyoruz. Çünkü elimizde daha fazla fraud olan veri yok.

2) Random Subsampling

Örnek sayısının az olduğu durumlarda kullanılan yöntemdir.

Her deneyi yaparken random olarak test setini belirliyoruz. Böylelikle modelin daha doğru çalışmasını sağlayabiliriz. * Bu şekilde n tane test yapıyoruz. * Hatayı her testing sonucundaki hatanın ortalaması olarak alıyoruz.

3) K-Fold Cross Validation

Örnek sayısının az olduğu durumlarda kullanılan yöntemdir.

- Ayrı bir test datası kullanılmasına gerek kalmıyor.
- Bütün örneklerin hem test hemde trainde kullanılacağı bir yöntemdir.
- k değerini biz seçiyoruz. Örneğin 5.
- Ex1: Train, Train, Train, Train, Test
- Ex2: Train, Test, Train, Train, Train
- ...
- gibi 5 farklı yerden test sample ı seçiliyor.
- Leave one out Cross Validation
  - N - 1 örneğimiz train için, 1 tanesi test için kullanılıyor.
  - Çok az data olduğu zaman (Örn: 50 sample) kullanılıyor.

4) Bootstrap

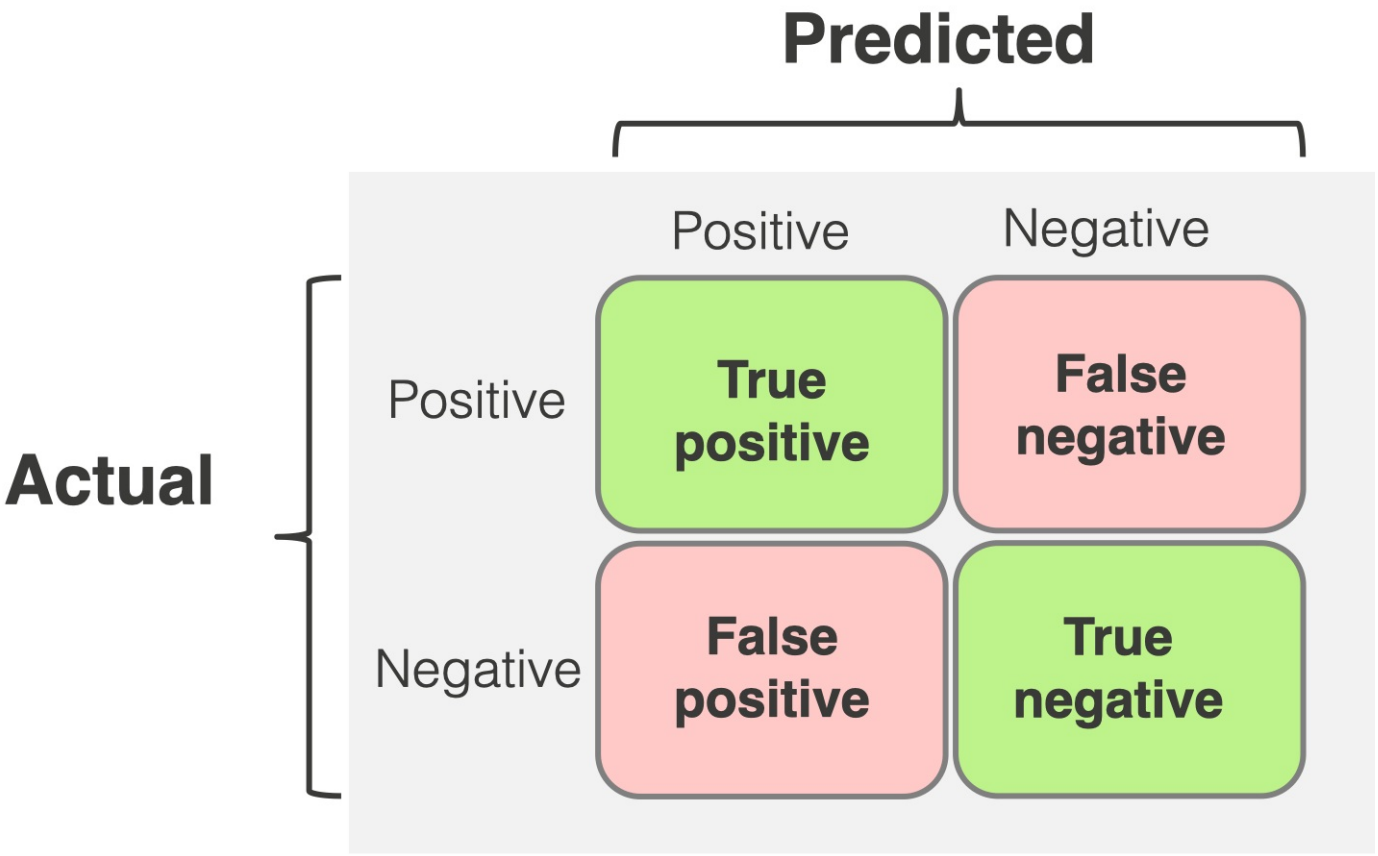
Randomly select N samples from dataset.

Model Performance Evaluation

- Performansını nasıl ölçeceğiz? (Performans Evaluation)
- Çok temel performans ölçme teknikleri var (Precision, Recall, Accuracy, F1-score, etc.)
- Literatürde benzer problemler için hangi metrik kullanılmış ona bir bakın. Buna göre daha rahat karar verebilirsiniz.

1) Confusion Matrix

Mutlaka çalışmalarınızda confusion matrix kullanın. Çünkü bu matrix size modelinizin ne kadar başarılı olduğunu gösterir.



- TP: True Positive
- TN: True Negative
- FP: False Positive
- FN: False Negative

Veriye göre hangi oranın azaltılması gerektiği değişebiliyor.

Positive: İlgilendiğimiz durum. * Mail spam tespitinde spam olanlar pozitif. * False positif: Mail spam değilken spam olarak işaretlenmesi mailin kaybedilmesine neden olur. Bunun azaltılması lazım. * Hastalık tespitinde hasta olunan durum pozitif. * Covid pozitif çıkması mesela. * False negative: Hastayken (kanser) hasta değil demek çok büyük bir problem. Bunun azaltılması gerekiyor.

2) Accuracy

Single Number Evaluation Metric: * Accuracy: (TP + TN) / (TP + TN + FP + FN)

Önemli Not: Accuracy nadir görülen durumlarda (fraud ile normal işlem) çok yüksek çıkmasına rağmen gerçekten modelin başarısı hakkında bilgi vermez. * Çünkü nadir durumlar çok az olduğu için modelin başarısı hakkında yanıltıcı olabilir. * Fraud detection gibi nadir durumlar için accuracy kullanılmamalıdır. * Imbalanced (unbalanced) data setlerde accuracy kullanılmamalıdır.

3) Precision

Pozitif olarak tahmin ettiklerinin içerisinde % kaçı positifti gerçekten? Bunu tahmin etmek için kullanılır. * Precision: TP / (TP + FP)

4) Recall

Gerçekten pozitif olanların % kaçını doğru olarak tahmin ettik? Bunu tahmin etmek için kullanılır. * Recall: TP / (TP + FN)

5) F1-Score

Ayrı ayrı precision ve recall değerlerini kullanmak yerine F1-score kullanılabilir. * Modelin genel başarısı hakkında bize bilgi verir. * F1-Score: 2 * (Precision * Recall) / (Precision + Recall)

Logistic Regression

Konu değiştiriyoruz.

- Bir sınıflandırma yöntemi.
- Regresyon yöntemi değil. Adından yanlış anlamayalım.
- Binary classification metodu
- Sigmoid fonksiyonu kullanılıyor.
- Verilen giriş değerleri için bilinmeyen örneğin hangi sınıfa ait olduğunun **olasılığını** verir.
- Genellikle **iki sınıflı** verileri sınıflandırmak için kullanılır.

Online Logistic Regression Algoritması; * Hatanın her adımda düzeltilmesi demek * Initialize weights [-0.5, 0.5] * repeat * next data point * update weights according to error * end * until covergance

Logistic regression içinde gradient descent kullanılıyor. Türev alıyoruz.

- 2 sınıfımız varsa bize bir olasılık sonucu verecek. Bu olasılığa bakarak hangi sınıfa ait olduğuna karar vereceğiz
- 2 den fazla sınıfımız varsa multinominal logistic regression oluyor.
  - Softmax(Zi) yöntemi kullanılır. Her sınıfın ayrı ayrı olasılığını veriyor.
  - Her örneğin bulunma olasılığını hesaplıyor.

- Toplam olasılık 1 oluyor.
- Softmax sonucu en büyük olasılık değerini alan sınıf, verinin gerçek sınıfıdır şeklinde yorumluyoruz.

6. Week - 4 November 2024 Monday

ANN: Artificial Neural Networks

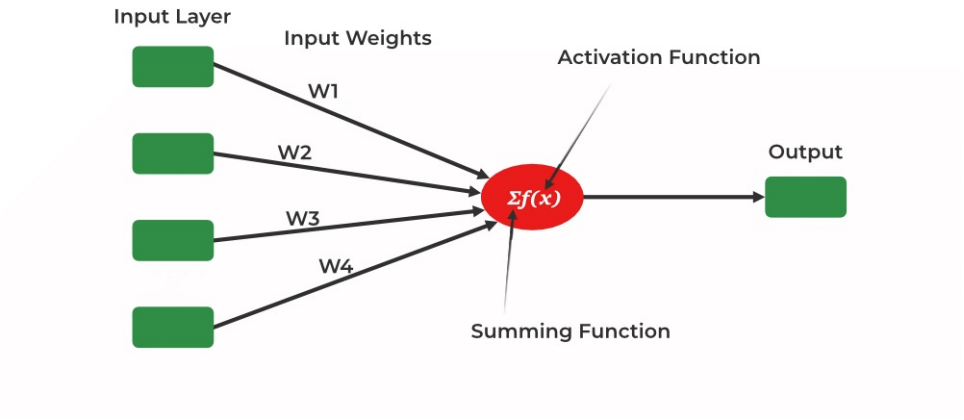
1958: Frank Rosenblatt introduces the Perceptron Perceptron: A single layer neural network

Eskiden paralel ve distributed computing yapılmaya çalışılıyordu. * Yapay nöron ağları beynin paaralel ve distributed yapısını taklit etmeye çalışıyor. * İnsan beyni 10 milyar nöron ve 60 trilyon sinaps içerir.

Single Layer Perceptron - On-Line Learning (Stochastic Gradient Descent)

- Her seferinde ağırlıkları update ederek hatayı minimize etmeye yani azaltmaya çalışıyoruz
- TODO: Ödevde SGD ile w0 ağırlığını update etmek gerekebilecek bir kısım olacak.
- Tek bir doğru çizilerek yapılabilen sınıflandırmalar için kullanılır.

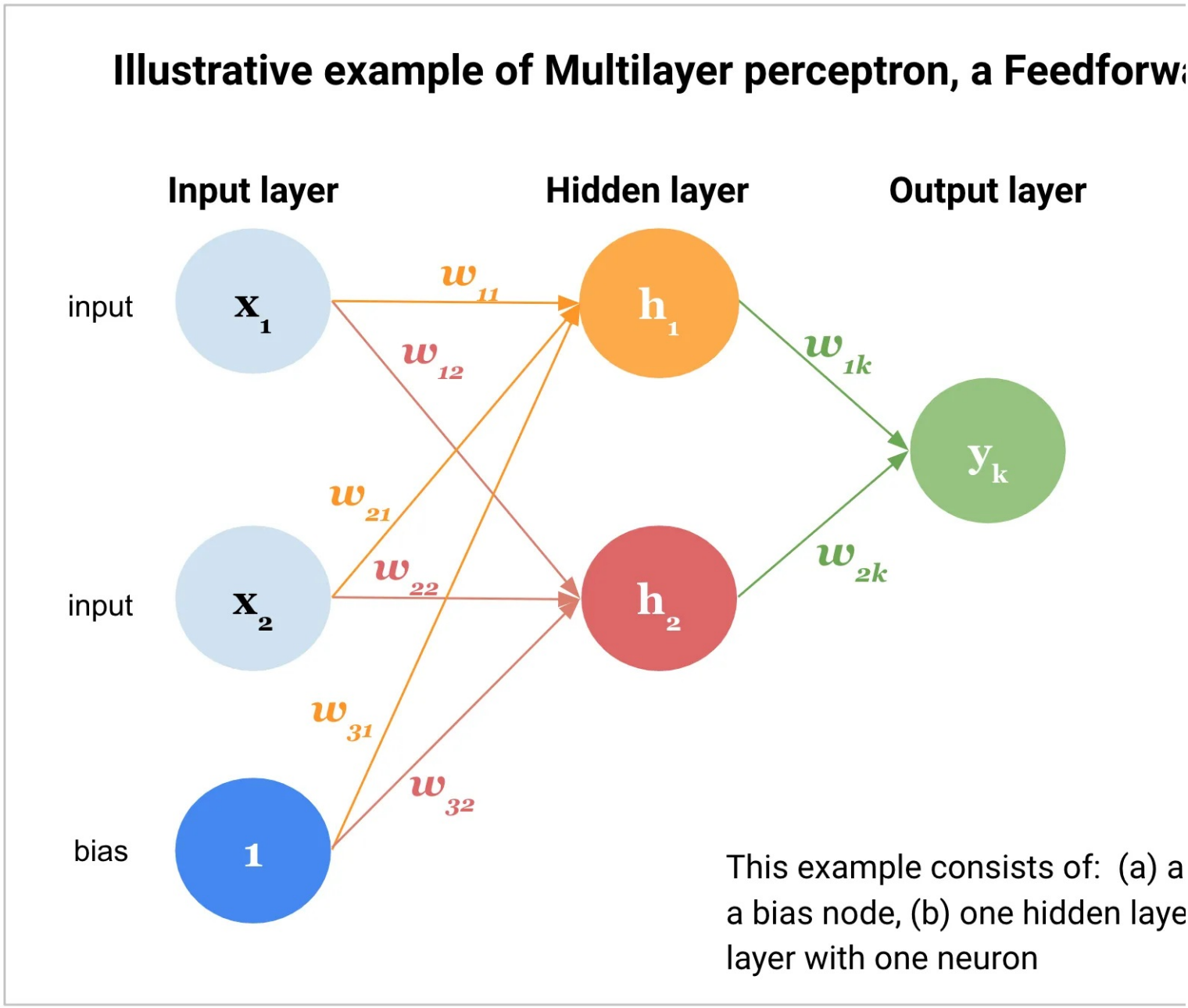
Source: https://media.geeksforgeeks.org/wp-content/uploads/20221219111343/Single-Layer-Perceptron.png



Multi Layer Perceptron - Feed Forward Neural Network (FFNN)

- Geri besleme yok
- Birden fazla doğru çizerek ifade edilebilecek sınıflandırmalar için kullanılır.
- İçeride birden fazla hidden layer bulunur. Bu şekilde input a uygun çıkış verilebilir.
- Hata update edilirken en sondaki layer update edilir, sonra bir önceki, sonra bir önceki şeklinde devam eder.

Source: https://aiml.com/wp-content/uploads/2022/06/Multilayer-perceptron-MLP.png



TODO: Bu hafta işlenen FFNN konusuna ait mavi renkli slayt ın 39. sayfasında olan örneği çözdük. Bundan belki sınavda soru gelebilir.

Batch Gradient Descent

- Yukarıdaki normal gradient descent ti.
- Batch olsaydı bütün örnekler için hata hesaplanıp sonra ağırlıklar güncellenecekti.

NOT: Mavi slayt üzerinden sayfa 50'ye kadar işledik.

7. Week - 11 November 2024 Monday

TODO: Vize hakkında;

- Vizemiz 9. hafta olacak.
- 25 Kasımda olacak

grafik verir; * hangisi over fitting olur sorusu sorar? * Buna göre yorumlama yapılabilir.

Support Vector Machines

- Linear ayrılabilen binary classification için tasarlanmıştır. 2 li sınıflandırma için tasarlanmıştır.
- Discriminative bir classifier dır.
- Hyper plane yani düz bir karar düzlemi çizilerek o düzleme göre sınıfları 2 ye ayırır.
  - Öyle bir yerden geçilelim ki hyper plane 2 sınıfın en yakın örneklerine de eşit uzaklıkta olsun.

SVM'i gösteren bir slayt: prepared by Martin Law * Veri Madenciliğinde kullanılan slaytlarla aynı slayt gibi.

Multi Class SVM

Normalda 2 li sınıflandırma problemi için oluşturulmuş yöntemin 2 den fazla sınıf için uyarlanmasıdır.

One Against-one SVM

A, B, C, D sınıfları olsun

training aşamasında; AB, AC, AD, BC, BD, CD ikili sınıflandırmalarını yapıyoruz. * Buradan çıkan ikili sınıflandırma sonuçlarından sonra A ve C çıktıysa mesela bu ikisi için tekrar SVM yaparak gerçek sonuca

ulaşıyoruz.

One Against-all SVM

- Denediğimiz zaman bu daha başarılı sonuç veriyor.
- Bunu kullanmayı düşünebiliriz.

A, B, C, D sınıfları olsun

training yaparken A yi hepsiyle işleme sokuyoruz. Eğer

Bayes Teoremi

- Sample ın sınıflara ait olma olasılıkları hesaplanır.
- Bunlardan hangisi daha yüksekse sample o sınıfa ait olarak varsayılır.
- Generative bir classifier dır.
- Bayes teoremi ile sınıflar arasındaki olasılıklar hesaplanır.
- Olasılıklar üzerinden hangi sınıfa ait olduğu belirlenir.

8. Week - 18 November 2024 Monday

Bayes Teoremi

Bayes teoreminden devam ediyoruz. Bugün için not alınmadı. Eski videolardan bakılarak not alınabilir.

9. Week - 25 November 2024 Monday

Bu hafta da ders işleyeceğiz.

K-Means Teoremi

K-Means teoreminden başlayarak devam ediyoruz. Bugün için not alınmadı. Eski videolardan bakılarak not alınabilir.

TODO: Yeterlilik sınavında K-Means complexity soruyorlarmış. Elif Hoca'nın K-Means notlarında var. #

Machine Learning Vize 1

Birkaç örnek bilgi veya bişeyler verir. Bu neden böyle oldu diye yorum sorusu sorar. Naive Bayes i matematiksel olarak sorabilir. Desicion Tree yi de matematiksel olarak sorabilir. Hesap makinesine ihtiyaç olmaz diye söyledi. Ama yine de getirilebilir.

10. Week - 2 December 2024 Monday

Vize sınavımız bu hafta olacak.

11. Week - 9 December 2024 Monday

Sınav Soruları

Soruların üzerinden geçiyoruz. Hepsine bakmadık. Hızlıca gözden geçirdik.

1. Soru

c. artar veya değişmez

2. Soru

a. Yanlış. Her zaman engellemez
b. Şehir, renk gibi şeyler. Distance based bir yöntem olduğu için kullanamayız.
c. Yanlış. İmbalanced verilerde yanlış.
d. Yanlış. Bağımsız olmalarına dayanır.

3. Soru

Hata 0 olur. Karar sınırı değişmeyeceğinden LOOCV sonucu değiştirmez.

Feature Selection

Feature Extraction

Mevcut özelliklerle aynı olmayan yeni bir özellik çıkarımı yapılabiliyor.

PCA: Principal Component Analysis

- Daha çok sinyal işlemede kullanılan bir yöntem.
- Bir ses sinyali tek boyutlu bir vektör. Burda neyi seçeceğimizi bilmediğimiz için özellikleri dönüştürerek aktarmak oldukça anlamlı hale geliyor.

Temel Özellikler: * Ortalama hesaplanır. * Varyans hesaplanır. * Eigen value ve Eigen vektör hesapları yapılır.

Yüz Tanıma

Test aşaması; * y -> kim? * ilk tüm feature ları traininde elde edilen mean lerden çıkar. * Sonrasında eigen vektör ve value ları elde et. * Sonrasında kim olduğuna karar vermek için aşağıdakilerden bir tanesini kullan. * Euclidean distance * K-NN * SVM

Yukarıdakileri uygulayan bir bitirme tezi: * Viola - Jones * https://en.wikipedia.org/wiki/Viola%E2%80%93Jones_object_detection_framework * https://www.cs.cmu.edu/~efros/courses/LBMV07/Papers/viola-cvpr-01.pdf

Reinforcement Learning

- Bir önceki aldığınız kararın sonucuna göre bir sonraki kararı alıyorsunuz.

- Çok başarılı bir yöntem değildi. Fakat yapay zeka alanında olan son gelişmelerle birlikte tekrar popüler hale geldi. (Deep Learning ile birlikte)

Markov Decision Process (MDP)

- MDP: Markov Decision Process, reinforcement learning için kullanılan temel modeldir.
  - Her alacağım karar bir önceki kararın sonucuna bağlıdır şeklinde bir modeldir.
  - Her kararın sonuçlarından en iyi sonucu almak için bir strateji belirlenir.
- Agent: İşi yapan, karar veren, öğrenen, karar alan birimdir.
- Environment: Agent'ın kararlarından etkilenen ortam.
- State: Agent'ın bulunduğu durum.
- Action: Agent'ın yapabileceği hareketler.
- Reward: Agent'ın aldığı ödül.

Ödev Açıklaması

- Bütün eğitmenli öğreticilerin performans analizinin yapılması ile alakalı bir ödev verilecek.
- Tüm gördüğümüz yöntemler verilen veri setine uygulanacak ve sonuçlar karşılaştırılacak.
- Veri setini hoca kendisi verecek

12. Week - 16 December 2024 Monday

13. Week - 23 December 2024 Monday

Hocanın özel bir durumu nedeniyle ders yapılamamıştır. Dersin telafisi yapılacaktır.

14. Week - 30 December 2024 Monday

Convolutional Neural Network (CNN) konusunu işliyoruz.

Convolutional Neural Network (CNN)

Karakteristik özellikleri çıkartabildiği için iyi bir yöntemdir. * En büyük problemimiz özelliklerin çıkartılmaıdır. CNN bunu ortadan kaldırıyor. * Ne kadar çok aynı türe ait özelliklerle eğitirseniz başarınız bir o kadar artar.

Farklı filtreler farklı özellikleri yakalayabiliyor. * Kullanılan çok sayıda filtre ile çok sayıda özellik çıkartılabilir. * Öğrenilebilir filtreler ile özellikler çıkartılabilir.

Convolution Layer

- Kernel ler ile (3x3, 5x5, 7x7) resim üzerinde gezinme yapılır.
- Padding: Resim kenarlarına 0 ekleyerek resmi büyütme işlemidir.
- Stride: Kernel in resim üzerinde gezinme adımıdır.
- weight: Kernel in ağırlıklarıdır. Yani aslında kernel içerisindeki değerler ağırlıklarımızı temsil eder.

Proje

Makale Hazırlarken

- Paper ı review edenler bi abstract a bir de introduction a bakarlar. İyiyse kalan kısımlara devam ederler.

Abstract

Abstract her şey bittikten sonra yazılır.

Introduction

Çalışmada şu şu yapılmıştır. * Kısa paper larda literatür review kısa geçersiniz ve introduction kısmında verirsiniz

Literature Review

Uzun paper larda ise 2-3 sayfa gibi literatür review anlatımı yapılır. Farklı bir başlık olarak. * Her çalışmayı bir paragrafta anlatman lazım. Bu paragraf aşağıdaki gibi olmalı. * Ne yapılmış? * Hangi yöntem? * Hangi veri kümesi? * Ölçme yöntemi? Diğer çalışmalarda %65 doğruluk çıktığını verirsen mesela * Sonuç? * Göçük altında kalan insan var mı için teknolojik çalışmalar * Doppler radar.

Sistem Tasarımı

- Çalışmayı özetleyecek muhakkak bir block diyagramı bulunmalı (CNN block diyagramı gibi bakınca her şey görünmeli)
- Ana işlem adımları görünsün
- Equation larınız olmalı
- Mutlaka grafiklerle desteklenmeli.

Deneysel Çalışmalar

- İyi analizler yapmanız
- Öğrenmenin tamamlandığından kesin emin olun.
- Gözle bakın. Yanlış olanlar neler? Bizim önerdiğimiz yöntemle ne oluyor? Olayı bi anlamalıyız.
- Farklı makine öğrenme yöntemleri için
  - Accuracy karşılaştırmaları olsun
  - Train, validation ve test için epoch loss grafiği gibi grafikler bulunsun.
  - başarılı olanlar neden başarılı olmuş anlayın?
  - başarısızlar neden başarısız oldu?
  - Analiz yapın.
- Deneysel sonuçlarda, çözemesenizde başarısızlığın nedenleri hakkında bilgiler vermeniz isteniyor.

Conculusion

Elde ettiğiniz bilgiler

Sunumu Hazırlarken

Sunum 10 dakika 10 slayt gibi olacak. * Amaç * Kapsam * Veri Seti tanıtımı * Sonuçlar * Çalıştığınız konu ile ilgili başarılı ve başarısız sonuçlar * Sunumlar 9:30 da başlayacak.

15. Week - 6 January 2025 Monday