Administrative Issues

• Registrations
• Access to Moodle (https://odtuclass.metu.edu.tr/)
• Project paper selection
  – https://docs.google.com/spreadsheets/d/1tzPHq_Vgu6gCwNyXJHGvqeA6pgU67H0nKYjqkisWfKc/edit?usp=sharing
  – Deadline: 29 March + 1-2 weeks
Deep learning and other approaches

Previously on CENG501!

Fig.: I. Goodfellow
So what *does* Deep (Machine) Learning bring/provide?

- **(Hierarchical) Compositionality**
  - Cascade of non-linear transformations
  - Multiple layers of representations

- **End-to-End Learning**
  - Learning (goal-driven) representations
  - Learning to feature extraction

- **Distributed Representations**
  - No single neuron “encodes” everything
  - Groups of neurons work together

*Slide Credit: Marc'Aurelio Ranzato, Yann LeCun*
Building A Complicated Function

Given a library of simple functions

$$f(x) = g_1(g_2(\ldots(g_n(x)\ldots)))$$

Idea 2: Compositions

- Deep Learning
- Grammar models
- Scattering transforms…
Deep Learning = Hierarchical Compositionality

Feature visualization of convolutional net trained on ImageNet from [Zeiler & Fergus 2013]

Slide Credit: Marc'Aurelio Ranzato, Yann LeCun

Previously on CENG501!
Distributed Representations Toy Example

Can we interpret each dimension?

\[(a)\] no pattern \[\bullet \bullet \bullet \bullet \bullet \]

\[\begin{array}{c}
| \hline
| \hline
| \hline
| \hline
| \hline
\end{array}\]

\[(b)\] no pattern \[\bullet \bullet \bullet \bullet \bullet \]

\[\begin{array}{c}
| \hline
| \hline
| \hline
| \hline
| \hline
\end{array}\]
It can make mistakes

<table>
<thead>
<tr>
<th>Gender Classifier</th>
<th>Darker Male</th>
<th>Darker Female</th>
<th>Lighter Male</th>
<th>Lighter Female</th>
<th>Largest Gap</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microsoft</td>
<td>94.0%</td>
<td>79.2%</td>
<td>100%</td>
<td>98.3%</td>
<td>20.8%</td>
</tr>
<tr>
<td>FACE**</td>
<td>99.3%</td>
<td>65.5%</td>
<td>99.2%</td>
<td>94.0%</td>
<td>33.8%</td>
</tr>
<tr>
<td>IBM</td>
<td>88.0%</td>
<td>65.3%</td>
<td>99.7%</td>
<td>92.9%</td>
<td>34.4%</td>
</tr>
</tbody>
</table>


Joy Buolamwini & Timnit Gebru
(A) **Cow: 0.99**, Pasture: 0.99, Grass: 0.99, No Person: 0.98, Mammal: 0.98

(B) No Person: 0.99, Water: 0.98, Beach: 0.97, Outdoors: 0.97, Mammal: 0.96, Water: 0.94, Seashore: 0.97, Beach: 0.94, Two: 0.94

Today

• Introduction to machine learning

• Towards deep learning
  • History of deep learning
  • Biological neuron
  • Artificial neuron
  • Perceptron learning
  • Linear classification/regression
  • Non-linear classification/regression
  • Multi-layer perceptrons
A CRASH COURSE ON MACHINE LEARNING

Facial images in this part of the lecture are from the JAFFE database: https://figshare.com/articles/journal_contribution/jaffe_desc_pdf/5245003
What is machine learning?

Finding “hidden” structures in data “automatically”.

Sinan Kalkan
**What is machine learning?**

Finding “hidden” structures in data “automatically”.

With supervision → Supervised learning

<table>
<thead>
<tr>
<th>Input</th>
<th>Label</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Image" /></td>
<td>Happy</td>
</tr>
<tr>
<td><img src="image2.png" alt="Image" /></td>
<td>Happy</td>
</tr>
<tr>
<td><img src="image3.png" alt="Image" /></td>
<td>Happy</td>
</tr>
<tr>
<td><img src="image4.png" alt="Image" /></td>
<td>Happy</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Input</th>
<th>Label</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image5.png" alt="Image" /></td>
<td>Unhappy</td>
</tr>
<tr>
<td><img src="image6.png" alt="Image" /></td>
<td>Unhappy</td>
</tr>
<tr>
<td><img src="image7.png" alt="Image" /></td>
<td>Unhappy</td>
</tr>
<tr>
<td><img src="image8.png" alt="Image" /></td>
<td>Unhappy</td>
</tr>
</tbody>
</table>

Sinan Kalkan
What is machine learning?
What is machine learning?

$x \in \mathcal{I}$

$y = f(x)$

$y \in \mathcal{O}$

$f(\ )$?
Machine learning pipeline

Training

Data with label (label: happy, unhappy)

- Size
- Texture
- Color
- Histogram of oriented gradients
- SIFT
- Etc.

Extract Features

“Learn”

Testing

Extract Features

Learned Models or Classifiers

Predict

Happy or Unhappy
General Approaches

**Generative**

- Learn a model for each class.

**Discriminative**

- Find separating line (in general: hyperplane)
- Fit a function to data (regression).
General Approaches (cont’d)

<table>
<thead>
<tr>
<th>Supervised</th>
<th>Unsupervised</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Instance</strong></td>
<td><strong>Extract Features</strong></td>
</tr>
<tr>
<td><strong>Label</strong></td>
<td><strong>Learn a model</strong></td>
</tr>
<tr>
<td>happy</td>
<td>e.g. SVM</td>
</tr>
<tr>
<td>unhappy</td>
<td></td>
</tr>
<tr>
<td>happy</td>
<td></td>
</tr>
<tr>
<td>unhappy</td>
<td></td>
</tr>
<tr>
<td>happy</td>
<td></td>
</tr>
<tr>
<td>unhappy</td>
<td></td>
</tr>
</tbody>
</table>

Spring 2021

Sinan Kalkan
# General Approaches (cont’d)

<table>
<thead>
<tr>
<th>Type</th>
<th>Generative</th>
<th>Discriminative</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Supervised</strong></td>
<td>Neural Networks</td>
<td>Support Vector Machines</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Neural Networks</td>
</tr>
<tr>
<td></td>
<td></td>
<td>K-Nearest Neighbors</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Decision Trees</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Random Forest</td>
</tr>
<tr>
<td><strong>Unsupervised</strong></td>
<td>Mixture Models</td>
<td>K-means</td>
</tr>
</tbody>
</table>

Outline for the Machine Learning Part

- Supervised Learning
- Unsupervised Learning
- Other Forms of Learning
- General Issues in Learning
- Evaluation
SUPERVISED LEARNING
Supervised Machine Learning

• Find the following mapping, given the training set \( \{x_i, y_i\}_{i=1}^{N} \):

\[
y = f(x)
\]

\[
f: \mathbb{X} \rightarrow \mathbb{Y}.
\]

• Targets can be for classification or regression

• Methods:
  – K Nearest Neighbors, Support Vector Machines, Decision Trees, Random Forest, Neural Networks

<table>
<thead>
<tr>
<th>Input</th>
<th>Target Class</th>
<th>Target Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="Happy face 1" /></td>
<td>Happy</td>
<td>0.95</td>
</tr>
<tr>
<td><img src="image2" alt="Happy face 2" /></td>
<td>Happy</td>
<td>0.80</td>
</tr>
<tr>
<td><img src="image3" alt="Happy face 3" /></td>
<td>Happy</td>
<td>0.65</td>
</tr>
<tr>
<td><img src="image4" alt="Happy face 4" /></td>
<td>Happy</td>
<td>0.88</td>
</tr>
</tbody>
</table>
Supervised Machine Learning Methods

K Nearest Neighbors

• Advantages:
  – No training
  – Simple

• Disadvantages:
  – Slow testing time (more efficient versions exist)
  – Needs a lot of memory

Fig: http://cs231n.github.io/classification/
Binary classification can be viewed as the task of separating classes in feature space:

\[ y = f(x) = \text{sign}(w^T x + b) \]
Supervised Machine Learning Methods

Support Vector Machines

• How many lines are there?
• Which one is optimal?
Supervised Machine Learning Methods

Support Vector Machines

- Distance from example $\mathbf{x}_i$ to the separator is $r = \frac{w^T \mathbf{x}_i + b}{\|w\|}$
- Examples closest to the hyperplane are support vectors.
- Margin $\rho$ of the separator is the distance between support vectors.

Maximum Margin Classification
- Maximizing the margin is good according to intuition.
- Implies that only support vectors matter; other training examples are ignorable.

Borrowed mostly from the slides of: ML Group, Uni of Texas at Austin; Mingyue Tan, Uni of British Columbia
Supervised Machine Learning Methods
Support Vector Machines

What we know:
• $w \cdot x^+ + b = +1$
• $w \cdot x^- + b = -1$
• $w \cdot (x^+ - x^-) = 2$

\[ \rho = \frac{(x^+ - x^-) \cdot w}{|w|} = \frac{2}{|w|} \]

$\rho =$ Margin Width

Borrowed mostly from the slides of: ML Group, Uni of Texas at Austin; Mingyue Tan, Uni of British Columbia
Supervised Machine Learning Methods
Support Vector Machines

- Then we can formulate the optimization problem:

  Find \( w \) and \( b \) that maximizes:

  \[
  \rho = \frac{2}{\|w\|'}
  \]

  such that: \( y_i(w^T x_i + b) \geq 1 \) for all \( (x_i, y_i), i=1..n \)

  which can be reformulated as:

  Find \( w \) and \( b \) that minimizes:

  \[
  \Phi(w) = \|w\|^2 = w^T w,
  \]

  such that: \( y_i(w^T x_i + b) \geq 1 \) for all \( (x_i, y_i), i=1..n \)

  or as the following in more compact form:

  \[
  \min_{w \in \mathbb{R}^d, b \in \mathbb{R}} \|w\|^2 - \sum_i \alpha_i (y_i (w^T x_i + b) - 1)
  \]
Supervised Machine Learning Methods

Support Vector Machines

Limitation: Not robust to noisy data:
- Solution: Allow some margin of error for each sample point

Find $w$ and $b$ that minimizes:

$$\Phi(w) = ||w||^2 = w^T w,$$

such that: $y_i(w^T x_i + b) \geq (1 - \epsilon_i)$ for all $(x_i, y_i), i=1..n$

and $\epsilon_1 + \cdots + \epsilon_n < C.$
Limitation: Not suitable for non-linearly separable data

- Solution: Expand the feature space using a “kernel”, a function calculating some similarity between samples as a dot product:
  \[ K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j) \]

- Example kernels:
  - Polynomial
    \[ K(x_i, x_j) = (x_i \cdot x_j + 1)^p \]
  - Radial-basis Function:
    \[ K(x_i, x_j) = \exp \left( -\gamma \sum_k (x_{ik} - x_{jk})^2 \right) \]

Borrowed mostly from the slides of: ML Group, Uni of Texas at Austin; Mingyue Tan, Uni of British Columbia
Supervised Machine Learning Methods

Decision Trees

![Decision Tree Diagram]

Supervised Machine Learning Methods

Decision Trees

- At each node $m$, we divide the space based on a simple rule: $x_i > \theta_m$
- Following the nodes, we reach leaves.
- Classification: Leaves are labels
- Regression: Leaves are numerical values (a function of local values)

Fig: Oya Celiktutan.
Supervised Machine Learning Methods
Random Forest

• Instead of a single strong classifier, have many weak classifiers and combine their answers.

Fig: https://towardsdatascience.com/understanding-random-forest-58381e0602d2

Tally: Six 1s and Three 0s
Prediction: 1

Take 9 trees
UNSUPERVISED LEARNING
Unsupervised Machine Learning

• Learning without labels
• Goal: Discover a better representation of data
• Uses:
  – Dimensionality reduction
  – Data visualization
  – Preparation for supervised learning
  – Clustering
• Methods:
  – Principle/Independent Component Analysis, k-Means/x-Means Clustering, Manifold learning methods, ...
Unsupervised Machine Learning Methods

Clustering with $k$-Means

$k$ shall be 2
Unsupervised Machine Learning Methods

Clustering with k-Means

Problem:

- Need to know $k$ beforehand.
Unsupervised Machine Learning Methods
Dimensionality Reduction

Principle Component Analysis

• Analyze variance in the data
• Find new set of axes such that along the first axis, variance is the highest; along the second, variance is the second highest etc.
OTHER FORMS OF LEARNING

- Self-supervised learning
- Weakly-supervised learning
- Zero-shot/Few-show learning
- Reinforcement Learning
- Meta-learning
- Life-long learning
Self-supervised Learning

- Form supervision from data itself by predicting one portion of data from another portion

Fig.: Doersch et al., 2015
Weakly-supervised Learning

E.g. weakly supervised object detection

There is at least one penguin in this photo
Few-shot/Zero-shot Learning

Fig: https://preparingforgre.com/item/42488
Reinforcement Learning

Fig: Richard Sutton
Meta Learning

- One learning module shaping/controlling another

Lifelong Learning

ISSUES IN MACHINE LEARNING
Model selection

\[ f(x) = Wx \]

\[ f(x) = W_1 x + W_2 x^2 \]

\[ f(x) = \sum_i W_i x_i^i \]

\[ f(x) = Wf_1(x) \]

\[ f(x) = W_1 f_1(x) + W_2 f_2(x)^2 \]

...
Model Complexity

Models range in their flexibility to fit arbitrary data

- **High bias, low variance**
  - Simple model
  - Small capacity may prevent it from representing all structure in data

- **Low bias, high variance**
  - Complex model
  - Large capacity may allow it to memorize data and fail to capture regularities

Slide Credit: Michael Mozer
Bias-Variance Dilemma

![Graph showing the relationship between model complexity and error on test set, illustrating the trade-off between bias and variance. The graph depicts two curves: one for bias and another for variance. The point of optimum model complexity is where the total error is minimized.](image-url)

- **Error on Test Set**
- **Model Complexity**
- **Underfit**
- **Overfit**

Slide Credit: Michael Mozer
EVALUATION
How to evaluate performance

• Hold-out method:
Cross-validation

- K-fold cross validation

![Diagram of K-fold cross-validation process](https://scikit-learn.org/stable/modules/cross_validation.html)
Cross-validation

Exhaustive validation:

– Leave-p-out (LPO) cross validation
  • One iteration: Use p samples as the validation set and others as training set
  • Repeat this for all possible combinations of p samples.

– Leave-one-out (LOO) is LPO with p=1
Cross-validation

• Group/subject-based validation: Determine the folds based on group/subject information

• Leave one subject out (LOSO) cross validation

• Divide dataset into $k$ subsets with respect to subject identifiers

Training and test sets contain the same subjects

Training and test sets do NOT contain the same subjects
Evaluation Metrics

Classification

<table>
<thead>
<tr>
<th></th>
<th>actual class</th>
</tr>
</thead>
<tbody>
<tr>
<td>positive</td>
<td></td>
</tr>
<tr>
<td>true positives</td>
<td></td>
</tr>
<tr>
<td>(TP)</td>
<td></td>
</tr>
<tr>
<td>false positives</td>
<td></td>
</tr>
<tr>
<td>(FP)</td>
<td></td>
</tr>
<tr>
<td>negative</td>
<td></td>
</tr>
<tr>
<td>false negatives</td>
<td></td>
</tr>
<tr>
<td>(FN)</td>
<td></td>
</tr>
<tr>
<td>true negatives</td>
<td></td>
</tr>
<tr>
<td>(TN)</td>
<td></td>
</tr>
</tbody>
</table>

accuracy = \frac{TP + TN}{TP + FP + FN + TN}

Credit: Oya Celiktutan
Evaluation Metrics

Classification

• Always try to quantify the predictions that have not been made.
• Recall & F-measure are frequently used

\[
\begin{align*}
TP \ Rate \ (Recall) &= \frac{TP}{TP+FN} \\
FP \ Rate &= \frac{FP}{FP+TN} \\
Precision &= \frac{TP}{TP+FP} \\
F - measure &= 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}
\end{align*}
\]

Credit: Oya Celiktutan
Evaluation Metrics

Regression

Let $y_k$ and $\hat{y}_k$ be the ground-truth and predicted labels ($N$: number of predictions).

- **Mean Squared Error:**
  \[
  MSE = \frac{1}{N} \sum_{k=1}^{N} (y_k - \hat{y}_k)^2
  \]

- **Correlation:**
  \[
  COR = \frac{\sum_{k=1}^{N} (y_k - \mu_{y_k})(\hat{y}_k - \mu_{\hat{y}_k})}{\sqrt{\sum_{k=1}^{N} (y_k - \mu_{y_k})^2 \sum_{k=1}^{N} (\hat{y}_k - \mu_{\hat{y}_k})^2}}
  \]
  where $\mu_{y_k}$ and $\mu_{\hat{y}_k}$ are the sample means.

- **Coefficient of determination:**
  \[
  R^2 = 1 - \frac{\sum_{k=1}^{N} (y_k - \hat{y})^2}{\sum_{k=1}^{N} (y_k - \mu_{y_k})^2}
  \]

Credit: Oya Celiktutan
Training Vs. Test Set Error

Error

Optimum Model Complexity

Test Set

Training Set

Slide Credit: Michael Mozer
Facial Emotion Recognition

- Using SVM and MLP on Google Colab + sci-kit learn:
  https://colab.research.google.com/drive/1mZimbXBAJIqgl-04phHxXRYku6EvgUOw