

Computer Electronics

Lecture 10: Presentation of the lab
project: the k-nearest neighbors
algorithm

The timer design: what we have learned

- Creating a hardware peripheral for the iob-soc system using Verilog and the iob-lib macros and python scripts
- Creating the software driver for the peripheral using the C language and the iob-lib python scripts
- Creating documentation for the peripheral
 - Not fully explained
 - Visit the ug.pdf file

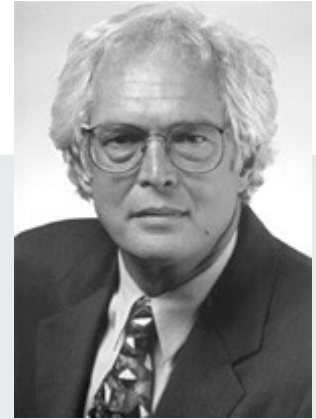
The User Guide

- Title page: ug.tex, version.txt, shortHash.txt
 - Visit and/or grep
- Table of contents, list of figures and tables
- Introduction: intro.tex
- Symbol: symb.odg
- Features: features.tex
- Benefits: benefits.tex
- Deliverables: deliverables.tex
- Block diagram: bdd.tex, bd.odg
- Table of block descriptions: bd_tab.tex, iob-timer.v

The User Guide (continued)

- Synthesis parameters: sp_tab.tex, iob-timer.v
- Interface signals: iob-timer.v, cpu_nat_s_if.v, gen_if.v
- Table of registers: timer_sw_reg.v
- Tables of FPGA results: ir.tex, alt_results.tex, xil_results.tex

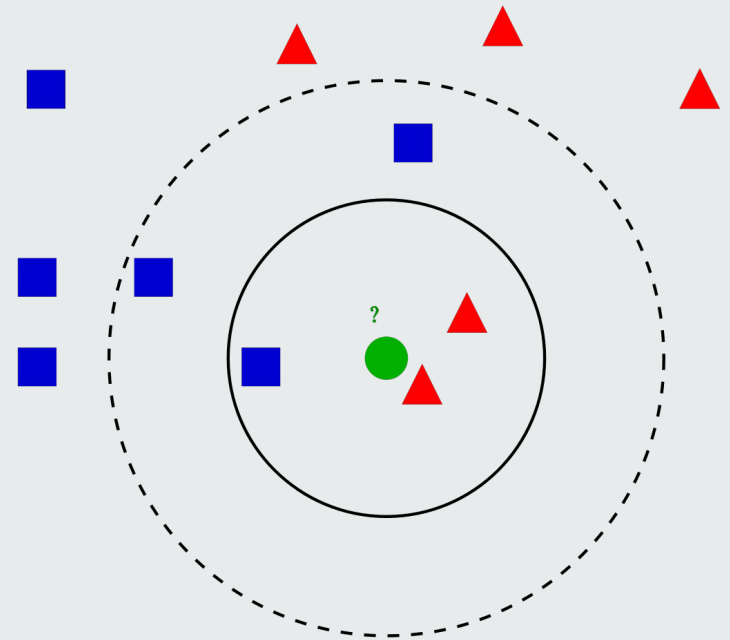
The k-nearest neighbors (KNN) algorithm



- Machine learning algorithm
- Invented by Thomas M. Cover, Stanford University
- Useful in pattern recognition
- Several application in biomedical sciences and technology

The KNN algorithm

- Used for classifying data items
- Requires dataset of already classified data: the labeled dataset
- Will classify a set of data items: the test set



The KNN algorithm steps

- 1) Set K , the number of neighbors
- 2) For each test item
- 3) Compute the “distance” to all dataset items
- 4) Keep the k -nearest neighbours
- 5) Set the label of the test item as the most common label among the k neighbors

